

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО”
ФАКУЛЬТЕТ ІНФОРМАТИКИ ТА ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ
Кафедра автоматизованих систем обробки інформації і управління

«До захисту допущено»

В.о. завідувача кафедри

_____ О.А.Павлов
(підпис) (ініціали, прізвище)

“ _____ ” _____ 2019 р.

Дипломний проект

на здобуття ступеня бакалавра

з напряму підготовки _____ 6.050103 «Програмна інженерія»

спеціальність _____ «Програмне забезпечення систем»

на тему: _____ Веб-сервіс для електронної видачі фото замовлень

Виконав: студент 4 курсу, групи ІІІ-52

_____ Куц Антон Віталійович

(прізвище, ім'я, по батькові)

_____ (підпис)

Керівник

_____ ас. каф. АСОІУ Недашківський Євген Анатолійович

(посада, вчене звання, науковий ступінь, прізвище та ініціали)

_____ (підпис)

**Консультант з
графічної
документації**

_____ доц. к.т.н. Ліщук К.І.

(посада, вчене звання, науковий ступінь, прізвище та ініціали)

_____ (підпис)

Рецензент

_____ доц.каф.ОТ,к.т.н.,доц. Болдак А.О.

(посада, вчене звання, науковий ступінь, прізвище та ініціали)

_____ (підпис)

Засвідчую, що у цьому
дипломному проекті немає
запозичень з праць інших
авторів без відповідних
посилань.

Студент _____
(підпис)

Київ – 2019 року

**Національний технічний університет України
“Київський політехнічний інститут ім. І.Сікорського”**

Факультет (інститут) _____ Інформатики та обчислювальної техніки
(повна назва)

Кафедра _____ автоматизованих систем обробки інформації і управління
(повна назва)

Рівень вищої освіти – перший (бакалаврський)

Напрямок підготовки _____ 6.050103 «Програмна інженерія»

ЗАТВЕРДЖУЮ

В.о. завідувача кафедри

_____ О.А.Павлов
(підпис) (ініціали, прізвище)

“ _____ ” _____ 2019 р.

**ЗАВДАННЯ
НА ДИПЛОМНИЙ ПРОЕКТ СТУДЕНТУ**

Куш Антон Віталійович

(прізвище, ім'я, по батькові)

1. Тема проекту Веб-сервіс для електронної видачі фото замовлень

керівник проекту Недашківський Євген Анатолійович, ас. кафедри АСОІУ

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджена наказом по університету від « _____ » _____ 2019 р. № _____

2. Термін подання студентом проекту « _____ » _____ 2019 року

3. Вихідні дані до проекту

Технічне завдання

4. Зміст пояснювальної записки

1) Аналіз вимог до програмного забезпечення

2) Моделювання та конструювання програмного забезпечення

3) Аналіз якості тестування та програмного забезпечення

4) Впровадження та супровід програмного забезпечення

5. Перелік графічного матеріалу (із зазначенням обов'язкових кресленників, плакатів, презентацій тощо)

1) Схема структурна варіантів використання програмного забезпечення

2) Схема структурна баз даних програмного забезпечення

3) Креслення екранних форм

6. Консультанти розділів проекту

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання « » 2019 року

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів виконання дипломного проекту	Термін виконання етапів проекту	Примітка
1	<i>Вивчення предметної області</i>		
2	<i>Аналіз існуючих методів розв'язання задачі</i>		
3	<i>Постановка та формалізація задачі</i>		
4	<i>Аналіз вимог до програмного забезпечення</i>		
5	<i>Моделювання програмного забезпечення</i>		
6	<i>Оформлення пояснювальної записки</i>		
7	<i>Подання ДП на попередній захист</i>		
8	<i>Подання ДП рецензенту</i>		
9	<i>Подання ДП на основний захист</i>		

Студент

(підпис)

Куш А.В.

Керівник проекту

(підпис)

Недашківський Є.А.

[illegible]

АНОТАЦІЯ

Пояснювальна записка дипломного проекту складається з чотирьох розділів, містить 28 таблиць та 9 рисунків – загалом 65 сторінок.

Об'єкт дослідження: веб-сервіс для електронної видачі фото замовлень із системою реєстрації та авторизації та можливістю інтеграції та синхронізації з Dropbox.

Мета дипломного проекту: створення єдиної платформи для фотографів та їх клієнтів, яка буде автоматизованою системою для електронної видачі фото замовлень.

У першому розділі було проаналізовано предметну область, розроблено функціональні вимоги та варіанти використання.

У другому розділі було розроблено та реалізовано архітектуру веб-застосунку, розроблено та візуалізовано бізнес-процеси за допомогою нотації BPMN, описано класи, інтерфейси та їх методи.

У третьому розділі було описано процес тестування програмного забезпечення.

У четвертому розділі було описано процес розгортання програмного забезпечення та інструкцію користувача.

КЛЮЧОВІ СЛОВА: ФОТОПОСЛУГИ, ВИДАЧА ФОТО
ЗАМОВЛЕНЬ, ІНТЕГРАЦІЯ З DROPBOX

					КПІ.ІП-5208.045440.01.81	Арк.
						5
Змн.	Арк.	№ докум.	Підпис	Дата		

ABSTRACT

The explanatory note of the diploma project consists of four sections, containing 28 tables and 9 figures – a total of 65 pages.

Object of research: web service for issuing photo orders with a registration and authorization system and the ability to integrate and synchronize with Dropbox.

The purpose of the diploma project: creation of a single platform for photographers and their clients, which is an automated system for electronic issuance of photo orders.

In the first section, the subject area was analyzed, functional requirements and usage options were developed.

In the second section, the web application architecture was developed and implemented, the business processes were developed and visualized using the BPMN notation, classes, interfaces and their methods were described.

In the third section, the software testing process was described.

In the fourth section, the process of deploying the software and the user's manual were described.

KEYWORDS: PHOTO SERVICES, ISSUING PHOTO ORDERS, INTEGRATION WITH DROPBOX

					КПІ.ІП-5208.045440.01.81	Арк.
						6
Змн.	Арк.	№ докум.	Підпис	Дата		

Пояснювальна записка до дипломного проекту

на тему: *Веб-сервіс для електронної видачі фото замовлень*

Київ – 2019 року

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ	9
ВСТУП.....	10
1 АНАЛІЗ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	12
1.1. Загальні положення	12
1.2. Змістовний опис і аналіз предметної області.....	12
1.3. Аналіз успішних ІТ-проектів.....	155
1.3.1. Аналіз відомих програмних продуктів	155
1.4. Аналіз вимог до програмного забезпечення	17
1.4.1. Розроблення функціональних вимог.....	18
1.4.2. Розроблення нефункціональних вимог	30
1.4.3. Постановка комплексу завдань модулю	30
1.5. Висновки до розділу.....	31
2 МОДЕЛЮВАННЯ ТА КОНСТРУЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	32
2.1. Моделювання та аналіз програмного забезпечення	32
2.2. Архітектура програмного забезпечення.....	42
2.3. Конструювання програмного забезпечення	44
2.3. Аналіз безпеки даних.....	52
2.4. Висновки до розділу.....	52
3 АНАЛІЗ ЯКОСТІ ТА ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	53
3.1. Аналіз якості ПЗ	53
3.2. Опис процесів тестування	55
3.3. Опис контрольного прикладу	62
4 ВПРОВАДЖЕННЯ ТА СУПРОВІД ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	63
4.1. Розгортання програмного забезпечення	63
4.2. Робота з програмним забезпеченням	63
ВИСНОВКИ	64
ПЕРЕЛІК ПОСИЛАНЬ.....	65

**ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ,
СКОРОЧЕНЬ І ТЕРМІНІВ**

MVC – шаблон проектування програмного забезпечення.

EF – entity framework, об'єктно-орієнтована технологія для роботи з даними.

BRMN – нотація для візуалізації бізнес-процесів.

Access token – спеціальна випадково сгенерована строкова змінна, яка використовується для доступу до акаунту Dropbox.

IoC – inversion of control, принцип за яким кожен компонент системи повинен бути максимально ізольованим від інших компонентів.

					КПІ.ІП-5208.045440.01.81	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

ВСТУП

Розвиток інформатизації та інформаційних технологій з кожним роком змушує людей відмовлятися від печатаних матеріалів та очних зустрічей із замовником. На даний момент неможливо уявити роботу більшості галузей обслуговування без використання певного програмного забезпечення, яке полегшує роботу виконавців, так і надання роботи споживачам. Але виникає питання: яким чином передавати результати роботи. Для покращення взаємодії клієнта та фотографа зазвичай використовують загальні ресурси, такі як пошта або посилання на інші ресурси, які містять результат роботи фотографа. У другому випадку проблема в тому, що немає єдиної платформи, яка б працювала та задовольняла потреби як фотографів, так і їх клієнтів. Проблему єдиної платформи я і спробував вирішити у даному дипломному проекті.

Питання способу передачі даних здається тривіальним, але все ж таки потребує окремої уваги. Погодження про спосіб передачі кожного разу при виконанні роботи фотографом не є зручним і потребує окремих затрат у часі. Ідея єдиного сайту для фотографа та кінцевого споживача вирішує це питання. Відпадають всі невпевненості щодо платформи для надання результатів послуги.

Сучасні рішення пропонують або цільові власні сайти окремих фотографів, або ресурси занадто широкого використання, такі як email. Тому для ефективного та погодженого надання результатів роботи фотографа можна створити єдину систему, яка буде працювати в інтересах клієнтів та фотографів однаково та буде містити тільки ту інформацію, яка об'єднує цих людей.

Мета створення даної роботи – створення високоефективної платформи для видачі фотозамовлень, яка містить функціонал для налаштування доступу

					КПІ.ІП-5208.045440.01.81	Арк.
						10
Змн.	Арк.	№ докум.	Підпис	Дата		

конкретних клієнтів до конкретного контенту фотографа. Ця система містить серверну частину, яка виконує інтеграцію та синхронізацію контенту фотографа із його Dropbox акаунтом та інтуїтивно зрозумілий веб-інтерфейс, для керування доступу та перегляду фотографій.

Завданням даної роботи є розробка системи для видачі фотозамовлень та керуванням доступу клієнтів до робіт фотографів. Результатом роботи є платформа, яка може бути використана фотографами для електронної видачі фотозамовлень та клієнтами для отримання робіт фотографів.

					КПІ.ІП-5208.045440.01.81	Арк.
						11
Змн.	Арк.	№ докум.	Підпис	Дата		

1 АНАЛІЗ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

1.1 Загальні положення

На даний момент, електронна видача фотозамовлень значно переважає у порівнянні з іншим видом видачі (печатним). Це пов'язано із розвитком соціальних мереж, галузей обслуговування. Більшість фотографій та документів зберігаються у електронному вигляді.

Фотопослуги – це результат безпосередньої взаємодії виконавців і споживача, а також діяльність виконавців, спрямована на задоволення індивідуальних потреб споживачів (замовників) у фотозйомці. Фотопослуги на сьогоднішній день користуються величезним попитом, і все більше людей починають звертати увагу на якість їх надання.

Виходячи з того факту, що людина є соціальною істотою, з кожним роком з'являється все більший попит на якісні фотознімки, які виконуються або у спеціальному приміщенні з правильно встановленим світлом, або шляхом правильного вибору локації для фотографування. Більшість людей довіряють цю роботу професіональним фотографам.

Фотографи, у свою чергу, зазвичай обирають додаток, який включає у себе сервер на якому знаходяться всі зняті фотографії. Локальне зберігання на диску або флешці не розглядається, адже є великий ризик втрати усіх даних при поломці або втраті фізичного пристрою. Саме тому фотографи обирають cloud рішення. Також важливим пунктом у виборі додатку є наявність функції для синхронізації контенту із локальним вмістом папок. Тому при аналізі вимог до програмного забезпечення було взято до уваги загальні тенденції вибору додатків для збереження фотографій. Також важливою вимогою є захищеність даних клієнтів та наявність системи авторизації.

1.2 Змістовний опис і аналіз предметної області

					КПІ.ІП-5208.045440.01.81	Арк.
						12
Змн.	Арк.	№ докум.	Підпис	Дата		

Як було сказано у попередньому розділі, важливим етапом у аналізі предметної області є вибір додатку для збереження фотографій, з яким і буде йти робота у даному веб сайті.

Хмарне сховище – це модель хмарних обчислень, яка передбачає зберігання даних в Інтернеті за допомогою постачальника хмарних обчислювальних ресурсів, який надає сховище даних як сервіс і забезпечує управління ним. Хмарне сховище надається на вимогу в необхідному обсязі, оплачується за фактом використання і позбавляє від необхідності купувати власну інфраструктуру для зберігання даних і керувати нею. Це забезпечує гнучкість, глобальну масштабованість і надійність. Дані доступні в будь-який час і в будь-якому місці.

Додаток, який використовує хмарне сховище, має відповідати таким вимогам:

- мати онлайн сховище із достатнім об'ємом вільної пам'яті та забезпеченням надійності, доступності та безпечності даних;
- мати функцію синхронізації із локальним станом папок;
- бажано, щоб додаток був кросплатформенним.

Після пошуку додатків, які відповідають вищенаведеним вимогам було виділено декілька варіантів. Серед них: Dropbox, Microsoft OneDrive та Mega. Кожен з цих додатків має деякі переваги та недоліки у порівнянні одне з одним.

Властивості Microsoft OneDrive:

- дозволяє зберігати до 5 ГБ інформації безкоштовно;
- присутня інтеграція з Hotmail (Outlook.com), що дозволяє користувачу безпосередньо завантажувати фотографії з Hotmail і зберігати їх на OneDrive;

- стандартно встроєний у операційній системі Windows 10.

Недостатки Microsoft OneDrive:

- не є повністю кросплатформенною (немає клієнтського додатку для Linux).

Властивості Dropbox:

- дозволяє зберігати до 2 ГБ інформації безкоштовно;
- є повністю кросплатформенним.

Недостатки Dropbox:

- має безкоштовно 2 ГБ вільного об'єму пам'яті, але пропонує реферальну програму — кожному запрошеному користувачеві та тому, хто запросив, додають ще по +500 МБ. Таким шляхом можна збільшити обсяг ще на +16 ГБ, запрошуючи інших до Dropbox. Крім того, Dropbox додає додатковий простір за виконання окремих акцій з тестування, коментарів у соціальних мережах та інше.

Властивості Mega:

- дозволяє зберігати до 50 ГБ інформації безкоштовно;
- вихідний код клієнта Mega доступний на GitHub.

Недостатки Mega:

- наявне шифрування контенту акаунту, яке проходить у клієнтській частині сайту за рахунок javascript. Таким чином є ризик виникнення атаки MITM (man in the middle), суттю якої є підміна або відключення javascript коду.

Так як кожний додаток відповідає попереднім вимогам наведеним вище та з властивостей та недоліків кожного додатку не можна чітко вибрати один, було прийнято рішення провести додаткове дослідження на предмет

популярності додатків серед фотографів. В результаті цього дослідження було з'ясовано, що фотографи найчастіше збергають свої роботи у додатку Dropbox. Тому у якості додатку, яке містить хмарне сховище було обрано Dropbox. При роботі з цим додатком використовувався офіційний API з супутнім SDK для мови програмування C#.

1.3 Аналіз успішних IT-проектів

На сьогоднішній день існує безліч способів для електронної видачі фотозамовлень. Серед основних можна виділити надсилання email листа з прикріпленими фотографіями, створення цільового сайту фотографа, створення сайту, де фотографи можуть викладувати свої роботи і скидувати посилання на окремі галереї своїм клієнтам, генерування в додатку, який використовує хмарне сховище, посилання на папку та надсилання його в соціальній мережі або месенджері. Так як перше і останнє рішення не представляє собою окремий сервіс, їх можна відкинути та зробити висновок, що сервіси для електронної видачі фотозамовлень можна умовно поділити на дві групи: власні сайти фотографів, де їх клієнти можуть переглянути зроблені фотографії, та сайти зроблені для фотографів, щоб ділитися своїми роботами. Значним недоліком першої групи сайтів є їх обмеженість щодо одного фотографа. Якщо клієнт отримує послуги фотографування від двох окремих фотографів, йому обов'язково бути зареєстрованим на двох окремих сайтах. Друга група сайтів являє собою успішні IT-проекти, які користуються популярністю у фотографів. Розглянемо переваги та недоліки декількох сайтів цієї групи.

1.3.1 Аналіз відомих програмних продуктів

Найбільш популярні сервіси для електронної видачі та шерінгу фотографій:

Pixieset

Даний сервіс призначений для професійних фотографів для обміну, підтвердження, видачі та продажу фотографій в Інтернеті.

Даний веб додаток має приємний інтерфейс, можливість створення галерей, шляхом завантаження фотографій, створення посилання на окрему галерею, відправлення приватних галерей окремим клієнтам. Також даний сервіс має систему лайків для клієнтів. Галереї фотографів можуть також побачити всі користувачі, якщо ця галерея є публічною.

Недоліками цієї платформи є відсутність акаунту клієнта, зайшовши у який він може побачити надіслані йому галереї (видача галерей відбувається шляхом відправлення фотографій на вказані пошти), відсутність інтеграції з одним з популярних сервісів для збереження фотографій у хмарному сховищі (для розміщення кожної галереї на сайті, потрібно буде завантажити кожен галерею окремо).

Pic-Time

Цей сервіс є подібним до попереднього. Pic-Time також має можливість розміщення власних проектів та відправлення їх окремим вказаним клієнтам по пошті. Для клієнтів наявна система лайків, розповсюдження кожної окремої фотографії проекту такими сервісами як email та facebook та коментування фотографій. Крім того цей сайт містить графіки активності користувачів щодо проектів (ведеться статистика по кількості скачувань, переглядів, покупок тощо). Гарними нововведеннями цього сервісу є його настрююваність головної фотографії кожного проекту, вибір теми проекту, створення власного портфоліо на сайті.

Недоліками Pic-Time є:

- видача відбувається посиланнями на проекти (відправлення посилання поштою або вручну будь-яким іншим способом);
- відправляти проект можна максимум двом клієнтам одночасно;

– відсутність інтеграції з одним з популярних сервісів для збереження фотографій у хмарному сховищі.

ShootProof

Сайт працює за схемою галерей з фотографіями, які можна відправляти окремим клієнтам по пошті або робити публічними. Можна налаштувати режим доступу до галерей (публічні галереї або доступ за посиланням). Характерною рисою цього сайту є можливість налаштувати відправлення обраної галереї клієнтам, які відвідали дану галерею. У безкоштовному варіанті цей сервіс дозволяє завантажити лише 100 фотографій. Платні плани користувача дозволяють збільшити ліміт кількості завантажуваних фотографій аж до необмеженої кількості.

Недоліком є відсутність інтеграції з сервісами для збереження фотографій у хмарному сховищі.

Проаналізувавши найбільш популярні сервіси можна зробити висновок, що фотографу з уже наявним багажом фотографій, буде не комфортно розміщати фотографії на тому чи іншому сайті. До того ж не всіх фотографів приваблює опція публічної галереї. Крім того система, яка використовується для видачі фотозамовлень у проаналізованих сайтах (надсилання посилання на галереї поштою) руйнує всю ідею єдиної платформи для клієнтів та фотографів.

1.4 Аналіз вимог до програмного забезпечення

Програмне забезпечення для електронної видачі фотозамовлень має бути єдиною платформою для фотографів та їх клієнтів. Тому в системі повинні бути такі два типи користувачів: клієнти та фотографи.

Передбачається, що для доступу до всіх функцій сервісу користувач повинен бути авторизований у системі.

Фотограф має змогу інтегрувати свій локальний акаунт із власним Dropbox акаунтом, переглядати вміст власного Dropbox акаунту, надавати

доступ до певних папок певним клієнтам, переглядати папки та користувачів, які мають доступ до них, синхронізувати свій локальний акаунт із Dropbox акаунтом.

Клієнт має змогу переглядати папки, доступ до яких він має, в той же час і фотографів, які надали цей доступ, підтверджувати власність власного акаунту.

Функціонал програмного забезпечення повинен виглядати наступним чином:

- реєстрація як клієнтів так і фотографів;
- авторизація у систему як клієнтів так і фотографів;
- інтеграція локального акаунту фотографа із його Dropbox акаунтом;
- синхронізація локального акаунту фотографа із його Dropbox акаунтом;
- система надавання доступу до певних папок (для фотографів);
- відкликання доступу до певних папок (для фотографів);
- перегляд доступних папок та файлів (для клієнтів);
- підтвердження власності локального акаунту (для клієнтів).

1.4.1 Розроблення функціональних вимог

Розроблені варіанти використання для даного програмного продукту наведені у вигляді таблиць нижче.

Таблиця 1.1 – Варіант використання УС-1

Назва	Реєстрація фотографа.
Опис	Реєстрація фотографа у системі.
Учасники	Фотограф.
Постумови	Фотограф проходить реєстрацію та авторизується у системі.
Основний сценарій	<p>Відображається форма реєстрації із такими полями для вводу: ім'я, прізвище, нікнейм, електронна пошта, пароль, підтвердження паролю.</p> <p>Фотограф заповнює поля вводу та натискає кнопку «Register».</p> <p>Система відображає форму для входу у Dropbox акаунт із такими полями: електронна пошта, пароль.</p> <p>Заповнивши поля вводу, фотограф натискає кнопку підтвердження та погоджується з умовами доступу.</p> <p>Система авторизує фотографа.</p>
Розширення сценаріїв	<p>1) Введені дані не є валідними.</p> <p>1.1) Відображаються всі помилки, які пов'язані із невалідними введенними даними.</p>

Таблиця 1.2 – Варіант використання УС-2

					КПІ.ІП-5208.045440.01.81	Арк.
						19
Змн.	Арк.	№ докум.	Підпис	Дата		

Назва	Реєстрація клієнта.
Опис	Реєстрація клієнта у системі.
Учасники	Клієнт.
Постумови	Клієнт проходить реєстрацію та авторизується у системі.
Основний сценарій	Відображається форма реєстрації із такими полями для вводу: ім'я, прізвище, нікнейм, електронна пошта, пароль, підтвердження паролю. Клієнт заповнює поля вводу та натискає кнопку «Register». Система авторизує клієнта.
Розширення сценаріїв	1) Введені дані не є валідними. 1.1) Відображаються всі помилки, які пов'язані із невалідними введенними даними.

Таблиця 1.3 – Варіант використання УС-3

Назва	Авторизація користувача.
Опис	Авторизація користувача у системі.
Учасники	Клієнт або фотограф.
Постумови	Користувач авторизується у системі.

Продовження таблиці 1.3

Основний сценарій	Відображається форма входу із такими полями для вводу: електронна пошта, пароль. Користувач заповнює поля вводу та натискає кнопку підтвердження. Система авторизує користувача.
Розширення сценаріїв	1) Введені дані не є валідними. 1.1) Відображаються всі помилки, які пов'язані із невалідними введенними даними.

Таблиця 1.4 – Варіант використання UC-4

Назва	Перегляд доступних папок.
Опис	Клієнт має можливість переглядати папки, доступні від фотографів.
Учасники	Клієнт.
Передумови	Клієнт авторизований у системі.
Основний сценарій	При переході на вкладу перегляду доступних папок, система демонструє список доступних папок. Клієнт має можливість переходити у кожную папку та переглядати фотографії.
Розширення сценаріїв	1) Клієнт не підтвердив акаунт. 1.1) Відображається повідомлення про необхідність підтвердження акаунту для перегляду доступних папок.

Таблиця 1.5 – Варіант використання UC-5

Назва	Перегляд вмісту акаунту Dropbox.
Опис	Фотограф має можливість переглядати папки власного акаунту Dropbox.
Учасники	Фотограф.
Передумови	Фотограф авторизований у системі.
Основний сценарій	При переході на вкладу перегляду акаунту Dropbox, система демонструє вміст кореневої папки акаунту Dropbox. Фотограф має можливість переходити у кожен папку та переглядати фотографії.
Розширення сценаріїв	1) У певній папці відсутні фотографії. 1.1) Відображається повідомлення про відсутність файлів у поточній папці.

Таблиця 1.6 – Варіант використання UC-6

Назва	Надання доступу до папок.
Опис	Фотограф має можливість надавати доступ до папок певним користувачам системи.
Учасники	Фотограф.
Передумови	Фотограф авторизований у системі.
Постумови	Обрані клієнти отримують доступ до обраних папок.

Продовження таблиці 1.6

Основний сценарій	Користувач переходить на вкладу надання доступу, обирає клієнтів (шляхом введення їх поштової адреси) та папки, натискає кнопку підтвердження.
Розширення сценаріїв	1) Певні користувачі вже мають доступ до вибраних папок. 1.1) Відображається повідомлення про дублікатне надання доступу.

Таблиця 1.7 – Варіант використання УС-7

Назва	Перегляд наданих на доступ папок.
Опис	Фотограф має можливість переглядати папки, доступ на який він вже дав певним користувачам.
Учасники	Фотограф.
Передумови	Фотограф авторизований у системі.
Основний сценарій	При переході на вкладку перегляду наданих папок, система демонструє список папок разом із користувачами, які мають доступ до відповідних папок.

Таблиця 1.8 – Варіант використання УС-8

Назва	Відкликання доступу на папки.
Опис	Фотограф має можливість відкликати доступ на папки певним користувачам.
Учасники	Фотограф.

Продовження таблиці 1.8

Передумови	Фотограф авторизований у системі.
Постумови	Обрані клієнти втрачають можливість перегляду наданих фотографом папок.
Основний сценарій	При переході на вкладу перегляду наданих папок, користувач обирає користувача, якому він бажає заборонити доступ до певної папки та натискає кнопку «Revoke Grant». З'являється модальне вікно для повторного підтвердження відкликання доступу, фотограф підтверджує натискаючи кнопку «Yes».

Таблиця 1.9 – Варіант використання UC-9

Назва	Синхронізація акаунту Dropbox.
Опис	Фотограф має можливість синхронізувати локальний стан папок із акаунтом Dropbox.
Учасники	Фотограф.
Передумови	Фотограф авторизований у системі.
Післяумови	Локальний стан папок синхронізований із акаунтом Dropbox.
Основний сценарій	При переході на вкладу синхронізації акаунту Dropbox, користувач натискає на кнопку «Synchronize». Відбувається синхронізація вмісту папок. Відображається повідомлення про успішну синхронізацію та список папок, які були синхронізовані.

Продовження таблиці 1.9

Розширення сценаріїв	1) Синхронізація не є необхідною. 1.1) Відобража'ться повідомлення про відсутність необхідності синхронізації.
----------------------	---

Для спрощеного представлення варіантів використання було розроблено діаграму варіантів використання, яка наведена у графічних матеріалах у документі «Схема структурна варіантів використання».

Функціональні вимоги програмного забезпечення

Розроблені функціональні вимоги для даного програмного продукту наведені у вигляді таблиць нижче.

Таблиця 1.10 – Опис функціональної вимоги FREQ-1

Номер	FREQ-1
Назва	Реєстрація локального акаунту в системі
Опис	Система має реєструвати локальний акаунт користувача клієнта або фотографа у системі.

Таблиця 1.11 – Опис функціональної вимоги FREQ-2

Номер	FREQ-2
Назва	Підв'язування Dropbox акаунту фотографа з локальним акаунтом
Опис	Система має інтегрувати Dropbox акаунту фотографа для використання його вмісту у системі.

Таблиця 1.12 – Опис функціональної вимоги FREQ-3

					КПІ.ІП-5208.045440.01.81	Арк.
						25
Змн.	Арк.	№ докум.	Підпис	Дата		

Номер	FREQ-3
Назва	Авторизація у систему
Опис	Система має давати можливість авторизуватися у сисему клієнту або фотографу, які попередньо пройшли реєстрацію.

Таблиця 1.13 – Опис функціональної вимоги FREQ-4

Номер	FREQ-4
Назва	Підтвердження акаунту клієнта
Опис	Система має відсилати повідомлення на пошту щойно зареєстрованого клієнта для підтвердження його акаунту.

Таблиця 1.14 – Опис функціональної вимоги FREQ-5

Номер	FREQ-5
Назва	Перегляд списку папок
Опис	Система має давати можливість користувачам переглядати список папок.

Таблиця 1.15 – Опис функціональної вимоги FREQ-6

Номер	FREQ-6
Назва	Перегляд фотографій
Опис	Система має давати можливість користувачам переглядати фотографії.

Таблиця 1.16 – Опис функціональної вимоги FREQ-7

Номер	FREQ-7
Назва	Перехід між папками
Опис	Система має давати можливість користувачам переходити між папками, у тому числі з одного рівня на інший.

Таблиця 1.17 – Опис функціональної вимоги FREQ-8

Номер	FREQ-8
Назва	Обирання папок для надання доступу
Опис	Система має давати можливість зареєстрованим фотографам обирати папки серед усіх папок акаунту Dropbox, до яких вони хочуть, щоб певні клієнти мали доступ.

Таблиця 1.18 – Опис функціональної вимоги FREQ-9

Номер	FREQ-9
Назва	Пошук клієнтів для надання доступу
Опис	Система має давати можливість зареєстрованим фотографам обирати клієнтів, шляхом введення їх поштових адрес.

Таблиця 1.19 – Опис функціональної вимоги FREQ-10

Номер	FREQ-10
Назва	Отримання папок, до яких був наданий доступ певним клієнтам

Продовження таблиці 1.19

Опис	Система має давати можливість зареєстрованим фотографам переглядати клієнтів, доступ яким був наданий до відповідних папок.
------	---

Таблиця 1.20 – Опис функціональної вимоги FREQ-11

Номер	FREQ-11
Назва	Відкриття доступу до папки
Опис	Система має давати можливість зареєстрованим фотографам відкрити доступ до конкретних папок обраним клієнтам.

Таблиця 1.21 – Опис функціональної вимоги FREQ-12

Номер	FREQ-12
Назва	Синхронізація із Dropbox акаунтом
Опис	Система має давати можливість зареєстрованим фотографам синхронізувати локальний вміст Dropbox акаунту із хмарним сховищем Dropbox (мають оброблюватися такі варіанти змін у акаунті Dropbox: перейменування папок та файлів, видалення папок та файлів, додавання папок та файлів).

Таблиця 1.22 – Опис функціональної вимоги FREQ-13

Номер	FREQ-13
Назва	Отримання змін акаунту Dropbox

Продовження таблиці 1.22

Опис	Система має давати можливість отримувати зміни, які відбулися із акаунтом Dropbox зареєстрованого клієнта.
------	--

Таблиця 1.23 – Опис функціональної вимоги FREQ-14

Номер	FREQ-14
Назва	Отримання списку доступних папок
Опис	Система має давати можливість зареєстрованим клієнтам переглядати вміст папок доступних від різних фотографів.

Матриця трасування між варіантами використання та функціональними вимогами наведена на рисунку 1.1.

	FREQ-1: реєстрація локального акаунту в системі	FREQ-2: Під'язування Dropbox акаунту фотографів локальним акаунтом	FREQ-3: Авторизація у систему	FREQ-4: Підтвердження акаунту клієнта	FREQ-5: Перегляд списку папок	FREQ-6: Перегляд фотографій	FREQ-7: Перехід між папками	FREQ-8: Обирання папок для надання доступу	FREQ-9: Пошук клієнтів для надання доступу	FREQ-10: Отримання папок, до яких був наданий доступ певним клієнтам	FREQ-11: Відкликання доступу до папки	FREQ-12: Синхронізація із Dropbox акаунтом	FREQ-13: Отримання змін акаунту Dropbox	FREQ-14: Отримання списку доступних папок
UC-1: Реєстрація фотографів														
UC-2: Реєстрація клієнта														
UC-3: Авторизація користувача														
UC-4: Перегляд доступних папок														
UC-5: Перегляд вмісту акаунту Dropbox														
UC-6: Надання доступу до папок														
UC-7: Перегляд наданих на доступ папок														
UC-8: Відкликання доступу на папки														
UC-9: Синхронізація акаунту Dropbox														

Рисунок 1.1 – Матриця трасування функціональних вимог та варіантів використання

1.4.2 Розроблення нефункціональних вимог

Нефункціональні вимоги виглядають наступним чином:

- працювати на таких браузерах, як: Google Chrome, Opera, Mozilla Firefox, Edge;
- безпечність зберігання паролів у базі даних, шляхом їх хешування та використання функції KDF.

1.4.3 Постановка комплексу завдань модулю

Даний програмний продукт призначений для електронної видачі фотозамовлень, як єдина зручна платформа для клієнтів та фотографів.

Мета створення даної роботи – атоматизація процесу електронної видачі фотозамовлень за рахунок інтеграції акаунту фотографа зі сховищем його фотографій та інтуїтивного інтерфейсу для надання доступу до папок хмарного сховища. Результатом роботи є веб застосунок.

Для того, щоб проект повністю відповідав меті свого створення, він повинен виконувати ряд задач:

- реєстрація клієнта та фотографа;
- авторизація клієнта та фотографа;
- підтвердження акаунту клієнта;
- надання доступу до папок;
- відкликання доступу до папок;
- перегляд вмісту Dropbox акаунту;
- перегляд списку папок та клієнтів, які мають до них доступ;
- синхронізація Dropbox акаунту;
- перегляд доступних папок.

Веб сайт повинен коректно працювати для обох типів користувачів та виконувати задачі в залежності від типу авторизованого користувача.

					КПІ.ІП-5208.045440.01.81	Арк.
						30
Змн.	Арк.	№ докум.	Підпис	Дата		

1.5 Висновки до розділу

Із розвитком соціальних мереж та інформатизації послуг обслуговування з'явився великий попит на сервіси для електронної видачі фотозамовлень, які б виконували професіонали. До того ж найважливішою перевагою такого сервісу повинна стати легкість у інтеграції із вже зробленими фотографіями, які знаходяться на хмарному сховищі.

На даний момент існує ряд сервісів, які виконують подібні функції, але всі вони за основу видачі фотографій використовують email повідомлення. Даний спосіб має місце бути, але не є зручним, адже для відправлення або перегляду доступних фотографій потрібно одразу використовувати два додатки (сайт сервісу та сайт поштового ящика). До того ж жоден подібний сервіс не виконує інтеграцію фотографій із акаунтом хмарного сховища, який вже наявний у більшості професійних фотографів. Плюсом даного проекту також має стати його дієздатність при відсутності зв'язку із сервісом Dropbox.

У кінці кінців цей програмний продукт має відповідати вимогам, описаним у даному розділі та коректно оброблювати всі можливі передбачені функції.

					КПІ.ІП-5208.045440.01.81	Арк.
						31
Змн.	Арк.	№ докум.	Підпис	Дата		

2 МОДЕЛЮВАННЯ ТА КОНСТРУЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

2.1 Моделювання та аналіз програмного забезпечення

При моделюванні програмного забезпечення було вирішено використовувати BPMN нотацію для візуалізації бізнес-процесів. Загалом, клієнтська частина містить функціонал реєстрації та авторизації користувача у системі, керування доступом до папок, відображення вмісту папок.

Процес реєстрації клієнта у системі виглядає таким чином:

- клієнт вводить необхідні особисті дані у текстові поля та вибирає, що він реєструється, як «клієнт» (client);
- введені дані відправляються на сервер, який перевіряє, що клієнта з таким поштовим адресом ще немає в системі;
- у разі відсутності дублікатного користувача, сервер формує повідомлення для підтвердження акаунту та відсилає його на пошту клієнта, використовуючи сервіс SendGrid. Інакше відображається повідомлення про помилку;
- даний клієнт авторизується у системі.

Процес реєстрації фотографа у системі виглядає таким чином:

- фотограф вводить необхідні особисті дані у текстові поля та вибирає, що він реєструється, як «фотограф» (photographer);
- введені дані відправляються на сервер, який перевіряє, що фотографа з таким поштовим адресом ще немає в системі;
- користувач перенаправляється на форму інтеграції з його Dropbox акаунтом. Фотограф заповнює текстові поля пошти та пароллю для входу в Dropbox акаунт;

– користувачу пропонується погодитися із умовами програмного забезпечення щодо надання доступу до його Dropbox акаунту. Фотограф погоджується з умовами та натискає далі;

– у разі позитивної відповіді серверу, даний фотограф авторизується у системі, інакше відображається повідомлення про помилку.

Процес надання доступу до папок виглядає наступним чином:

– фотографу відображається список папок його Dropbox акаунту, текстове поле для введення пошти клієнта, кнопку підтвердження пошуку клієнта за введеною поштою та загальна кнопка підтвердження надання доступу;

– фотограф обирає папки зі списку, вводить пошти бажаних клієнтів та додає їх до списку клієнтів та натискає кнопку підтвердження. Дані відправляються на сервер, який перевіряє, чи мають обрані клієнти доступ до обраних папок. У разі відсутності помилок обраним клієнтам надається доступ до обраних папок, інакше виводяться повідомлення про повторне надання доступу.

Процес відкриття доступу до папок виглядає наступним чином:

– фотографу відображається список папок із відповідними клієнтами, які мають до них доступ;

– обравши певних клієнтів та натиснувши кнопку підтвердження, клієнтська частина відображає модальне вікно про підтвердження відкриття доступу до папок. Натиснувши «Так», дані відправляються на сервер та виконується відкриття доступу, інакше модальне вікно зникає та дані не відправляються на сервер.

Опис процесів при реєстрації клієнта:

– дані з форми реєстрації відправляються на сервер;

- сервер перевіряє, чи існує користувач за такою ж поштою в системі. Якщо існує, сервер відправляє на клієнтську частину відповідне повідомлення;
- відправлення повідомлення про підтвердження акаунту на електронну пошту клієнта, використовуючи сервіс SendGrid;
- сервер відправляє повідомлення про успішну авторизацію.

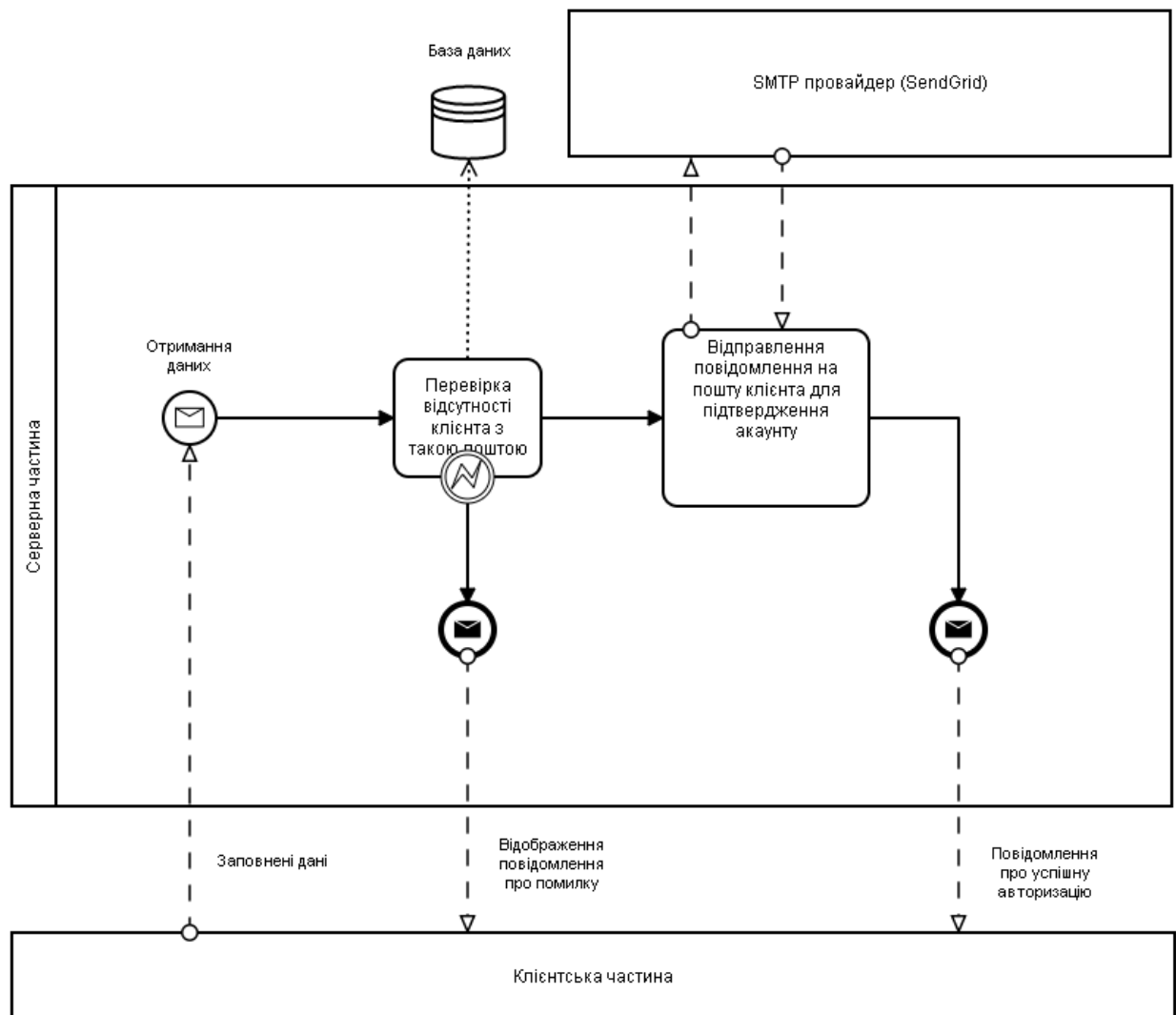


Рисунок 2.1 – Схема BPMN реєстрації клієнта

Опис процесів при реєстрації фотографа:

- дані з форми реєстрації відправляються на сервер;

- сервер перевіряє, чи існує користувач за такою ж поштою в системі. Якщо існує, сервер відправляє на клієнтську частину відповідне повідомлення;
- сервер перенаправляє користувача на сторінку авторизації у Dropbox, відправляючи змінні: посилання колбеку, секретний код, «Code» та «ConnectState». Фотограф заповнює форму для входу у Dropbox та натискає «Вхід». У разі коректного входу, фотограф погоджується із умовами доступу до Dropbox акаунту та натискає «Далі»;
- фотограф перенаправляється на адрес вказаний у посиланні колбеку, відправленому на попередньому кроці та сервер перевіряє відповідність та вірогідність змінних «Code» та «ConnectState». У разі помилки виводиться відповідне повідомлення;
- у разі позитивної відповіді сервер виконує інтеграцію Dropbox акаунту з базою даних;
- сервер відправляє повідомлення про успішну авторизацію.

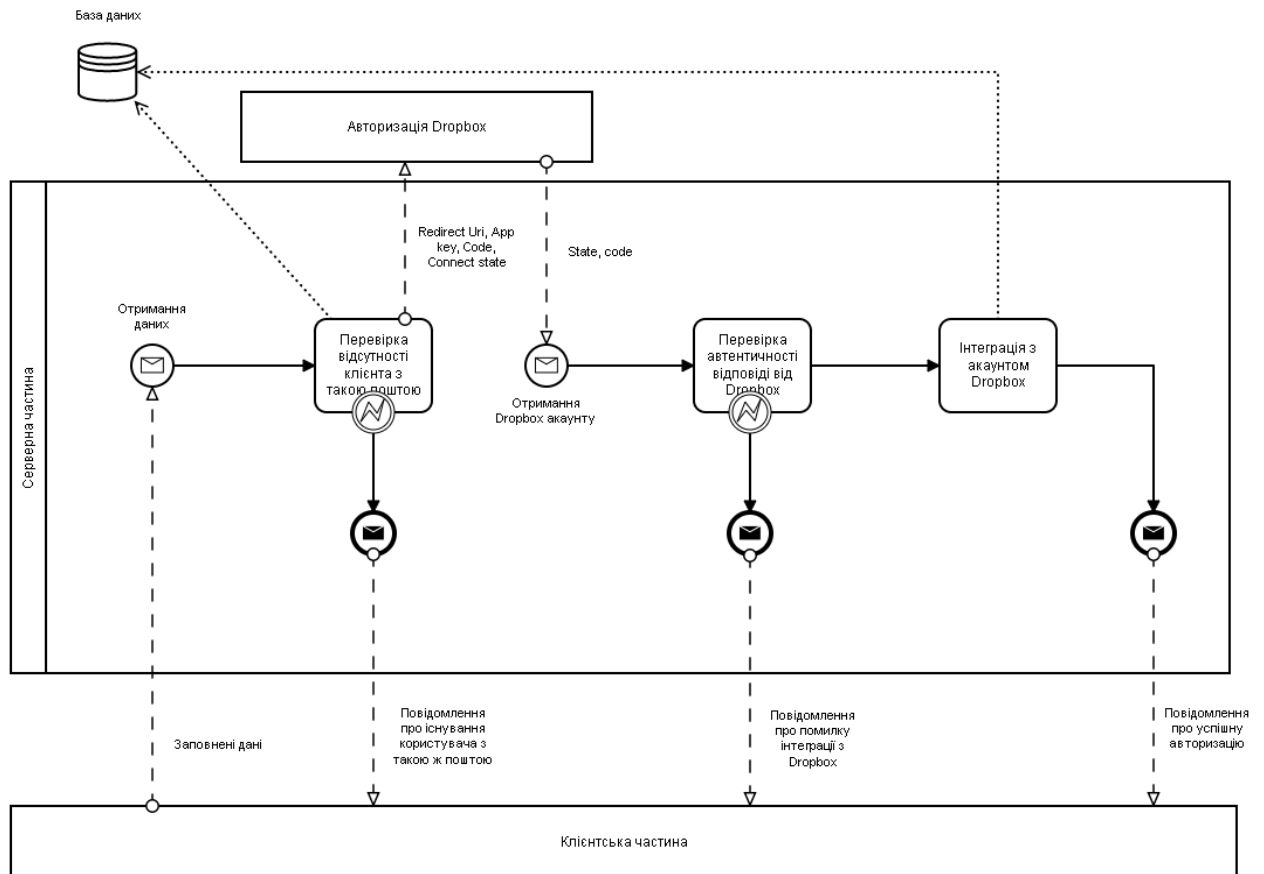


Рисунок 2.2 – Схема BPMN реєстрації фотографа

Опис процесів при авторизації користувача:

- клієнтська частина відправляє на сервер форму з поштою та паролем користувача;
- сервер перевіряє правильність комбінації пошти та паролю у базі даних. У разі некоректності пошти або паролю, сервер відправляє повідомлення про помилку авторизації;
- інакше сервер відправляє повідомлення про успішну авторизацію.

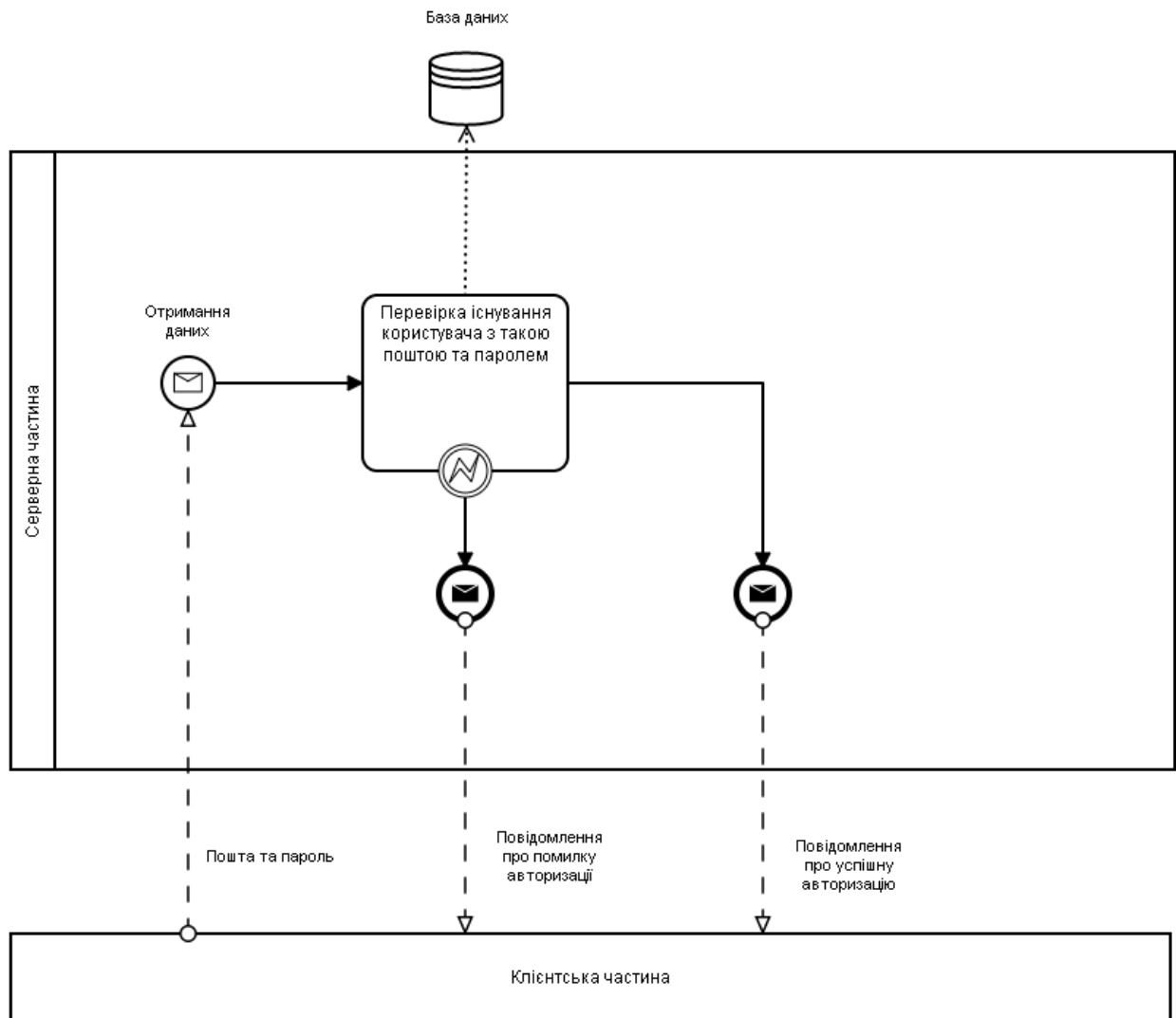


Рисунок 2.3 – Схема BPMN авторизації користувача

Опис процесів при наданні доступу до папок:

- клієнтська частина відправляє на сервер список папок та користувачів (їх поштових адрес);
- сервер перевіряє, чи авторизований даний фотограф у системі. Якщо ні, то сервер відправляє повідомлення про необхідність авторизації та перенаправляє користувача на форму входу;
- сервер перевіряє чи існують ці папки та користувачі у базі даних. У разі помилки відправляється відповідне повідомлення на клієнтську частину;

– сервер відправляє повідомлення про успішне надання доступу до папок.

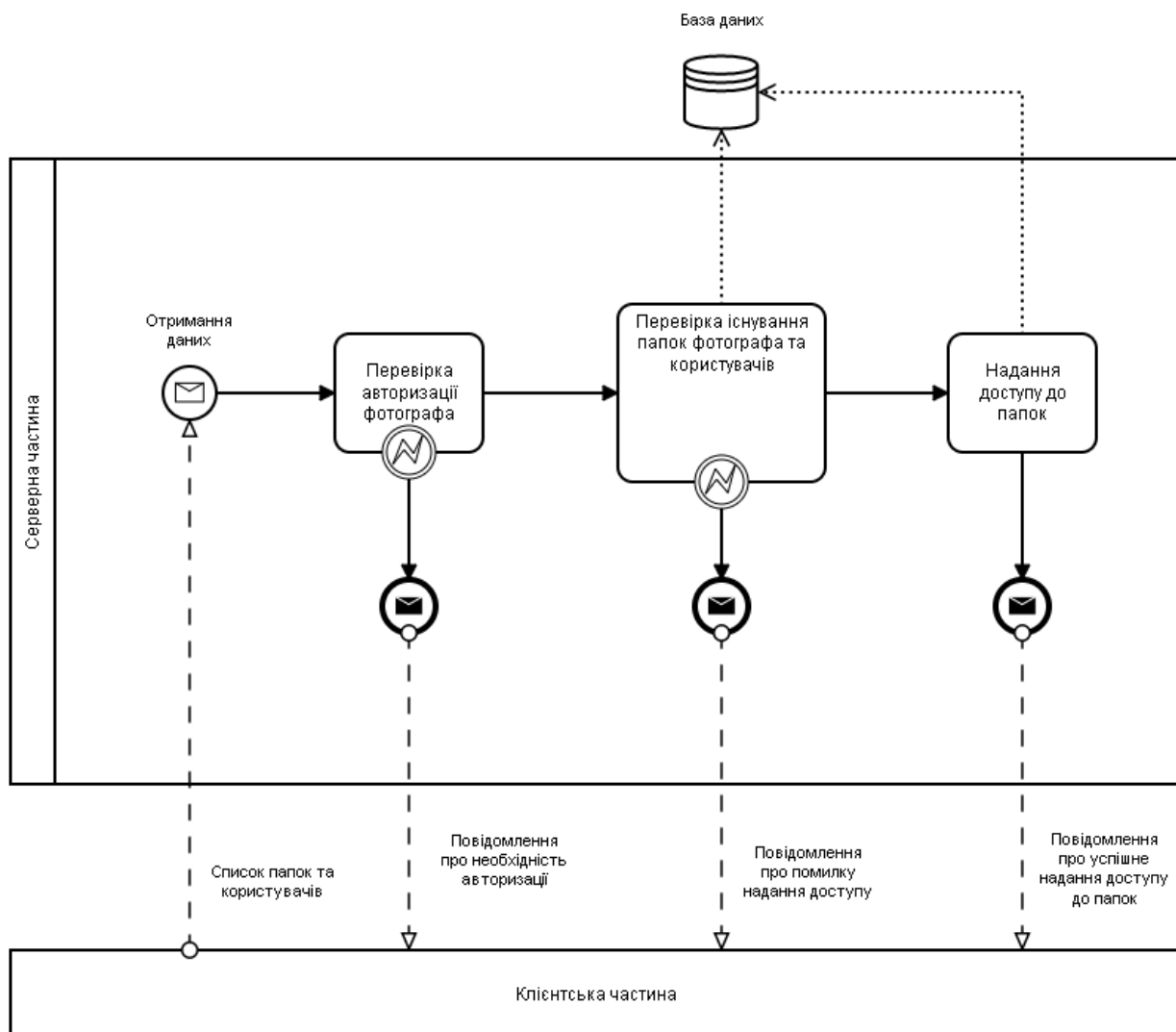


Рисунок 2.4 – Схема BPMN надання доступу до папок

Опис процесів при відкритті доступу до папок:

- клієнт обирає клієнтів та папки, доступ до яких хоче заборонити;
- клієнтська частина відображає модальне вікно підтвердження відкриття доступу.
- сервер перевіряє, чи авторизований даний фотограф у системі. Якщо ні, то сервер відправляє повідомлення про необхідність авторизації та перенаправляє користувача на форму входу;

- при натисканні «Так», клієнтська частина відправляє на сервер обрані списки папок та клієнтів (їх поштові адреси), що мають до них доступ;
- сервер перевіряє, що ці папки та користувачі існують у базі даних. У разі помилки відправляється відповідне повідомлення на клієнтську частину;
- серверна частина виконує операції відкриття доступу у базі даних та відправляє повідомлення про успішне виконання операції;
- клієнтська частина приймає успішне повідомлення та відображає відповідне модальне вікно.

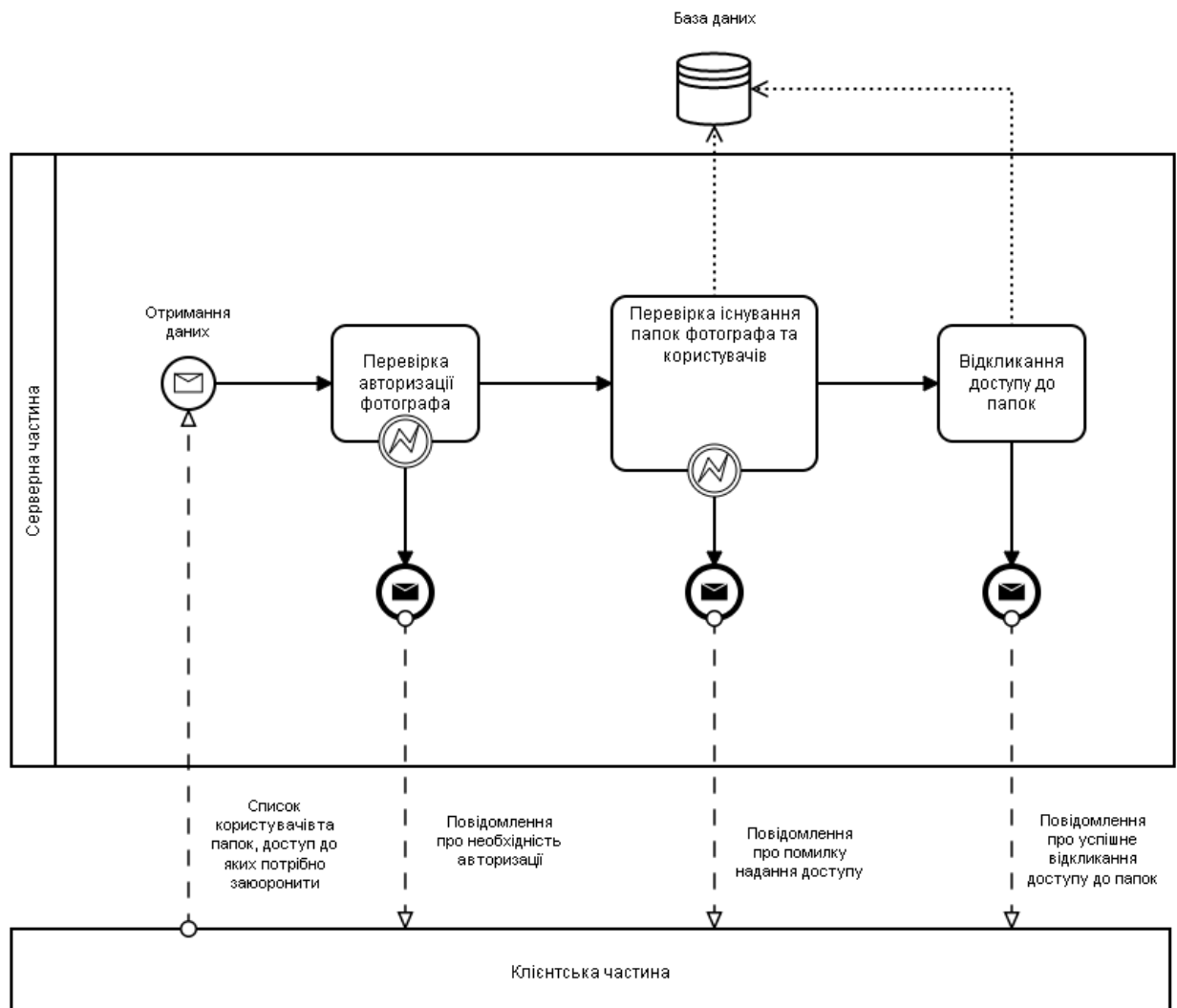


Рисунок 2.5 – Схема BPMN відкриття доступу до папок

Опис процесів при синхронізації акаунту фотографа з його Dropbox акаунтом:

- клієнтська частина відправляє на сервер запит на синхронізацію Dropbox акаунту із локальним станом папок;
- сервер перевіряє, чи авторизований даний фотограф у системі. Якщо ні, то сервер відправляє повідомлення про необхідність авторизації та перенаправляє користувача на форму входу;
- сервер виконує послідовні запити на Dropbox API для з'ясування змін у папках, використовуючи змінну «Cursor», яка записана у базі даних навпроти кожної папки у таблиці «PhotographerFolder», та виконується синхронізація стану папок у базі даних;
- сервер відправляє на клієнтську частину повідомлення, що містить список синхронізованих папок або повідомлення про те, що вміст папок не змінювався.

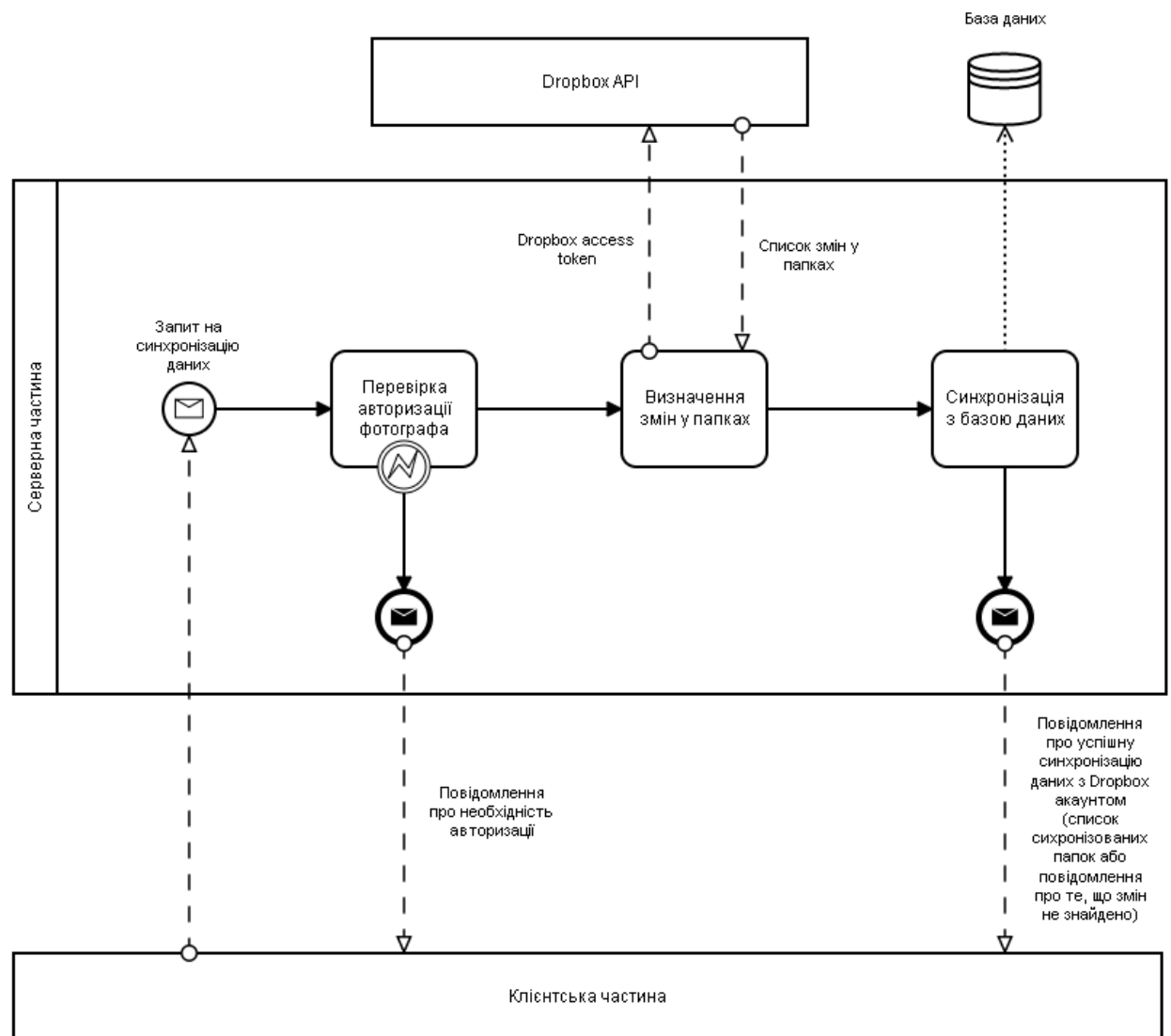


Рисунок 2.6 – Схема BPMN синхронізації з Dropbox

Опис процесів при відображенні клієнту списку доступних папок:

- клієнтська частина відправляє на сервер запит на отримання списку всіх папок доступних клієнту;
- сервер перевіряє, чи авторизований даний клієнт у системі. Якщо ні, то сервер відправляє повідомлення про необхідність авторизації та перенаправляє користувача на форму входу;
- сервер дістає з бази даних список всіх папок до яких даний клієнт має доступ та відправляє його на клієнтську частину.

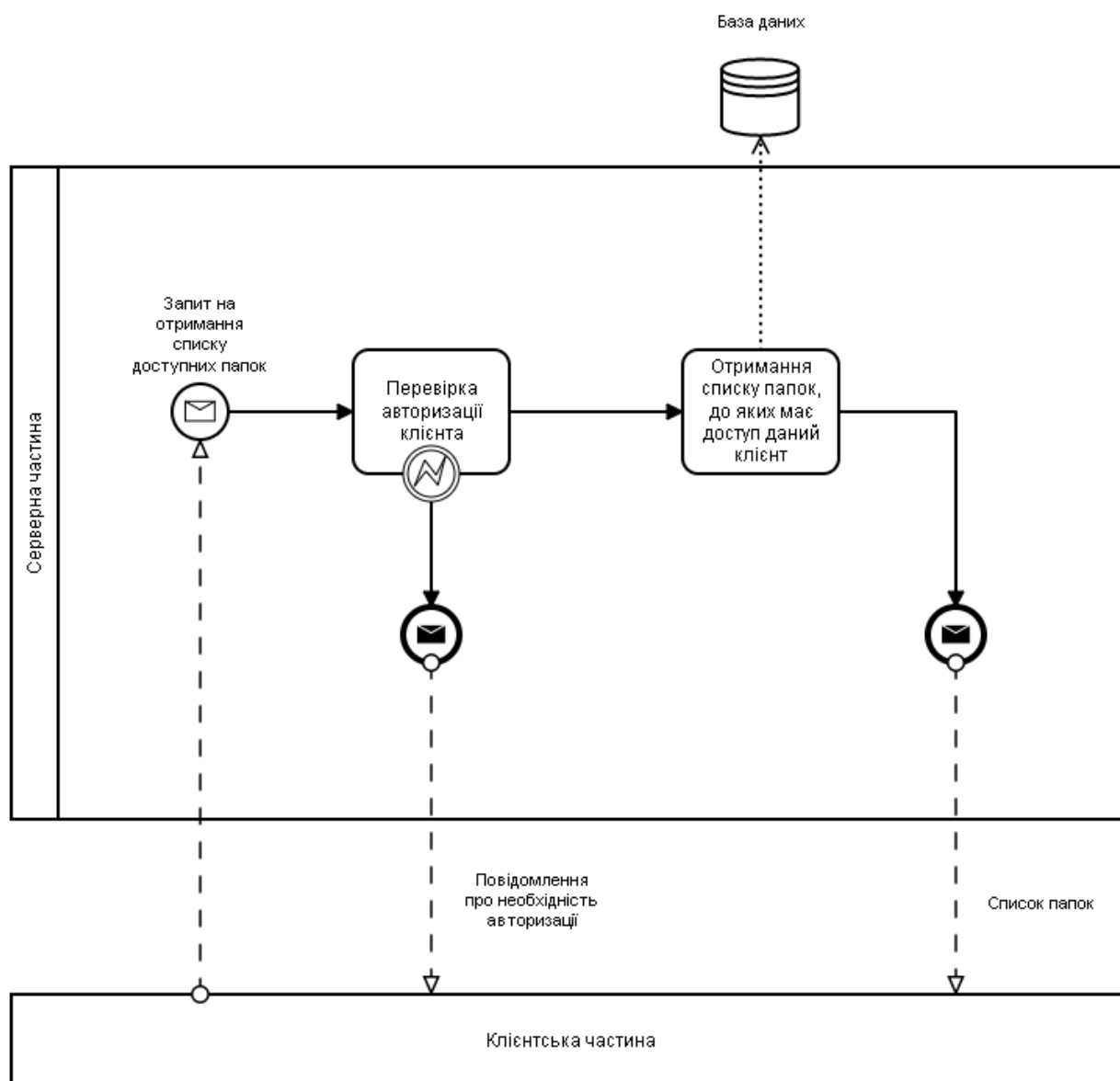


Рисунок 2.7 – Схема BPMN відображення доступних папок

2.2 Архітектура програмного забезпечення

Сервіс для електронної видачі фотозамовлень – веб застосунок, який був розроблений використовуючи інфраструктуру ASP.NET MVC та мову програмування C# для написання серверної частини програми. Для написання клієнтської частини та інтерфейсу сайту було застосовано розмітку HTML, таблиці стилів CSS та JavaScript бібліотеку JQuery, яка використовується для спрощеної взаємодії JavaScript та HTML.

Інфраструктура ASP.NET MVC за основу використовує шаблон MVC

Змн.	Арк.	№ докум.	Підпис	Дата

(Model-View-Controller), який з'явився ще в далекому 1978 році. Основними перевагами даного шаблону є те, що він розділяє технології на ряд рівні, що позитивно впливає на розширюваність, масштабованість модулів. Шаблон MVC розділяє програмний продукт на 3 частини: модель (частина, яка містить класи таблиць бази даних), представлення (частина, яка відображає дані отримані від моделі) та контролер (формує дані отримані від моделі та передає їх представленню на відображення). Найголовнішим плюсом даного шаблону є той факт, що кожна логічна частина може бути модифікована незалежно одне від одного. Таким чином відбувається відділення бізнес-логіки від її візуалізації, що сприяє повторному використанню коду.

Контролер містить інтерфейси сервісів, які містять всі можливі операції над даними. Для досягнення слабозв'язності коду було вирішено використовувати ІоС-контейнер Ninject. Зокрема у контролері використовується такий тип DI, як Constructor Injection. Контролер залежить від інтерфейсу (абстракції), а не від конкретної реалізації.

У якості бази даних було вибрано СУБД MS SQL Server. Для спрощеного проектування таблиць та зв'язків між ними було вирішено використовувати фреймворк Entity Framework та підхід Code First. Даний фреймворк дозволяє представляти таблиці бази даних у більш зрозумілому для розробника вигляді – у вигляді класів та об'єктів. Таким чином модель EF зберігає відомості про те, як класи і властивості додатку відображаються (від слова map) в таблиці, атрибути бази даних. Підхід Code First передбачає написання класів з властивостями, які потім за допомогою Entity Framework відображаються у таблиці у базі даних.

Для виконання задачі відправлення повідомлень на електронну пошту було вирішено використати сервіс SendGrid. SendGrid – це SMTP провайдер, який надає зовнішній поштовий сервер. Відповідно протокол за яким відбувається відправка – це SMTP (Simple Mail Transfer Protocol). Для

надсилання повідомлень на пошту було використано Web API даного сервісу із бібліотекою для мови програмування C#.

2.3. Конструювання програмного забезпечення

Опис інтерфейсів сервісів наведено в таблиці 2.1. Методи інтерфейсів та класів сервісів наведені у таблиці 2.2. Опис класів виключних ситуацій наведено в таблиці 2.3.

Таблиця 2.1 – Опис інтерфейсів сервісів.

Інтерфейс	Опис
IDropboxAccountService	Інтерфейс для взаємодії з контентом акаунту Dropbox та акаунтом Dropbox
IGrantAccessService	Інтерфейс для надання та відкликання доступу до папок
ILoginService	Інтерфейс для виконання авторизації користувача у системі
IProfileService	Інтерфейс для отримання користувачів сервісу та профілю окремого користувача
IRegistrationService	Інтерфейс для виконання реєстрації користувача у системі

Таблиця 2.2 – Опис методів інтерфейсів сервісів.

Інтерфейс/Клас	Метод	Опис
IDropboxAccount Service	GetDropboxAccountInfo(ApplicationUser currUser)	Отримання даних профілю акаунту Dropbox
IDropboxAccount Service	AddDropboxContentsToDB (ApplicationUser currUser)	Додавання вмісту акаунту Dropbox у базу даних
IDropboxAccount Service	GetFolderGrantedToClient(int folderId)	Отримання папки та її вмісту за ідентифікатором
IDropboxAccount Service	GetFolder(ApplicationUser currUser, string folderpath, int depth)	Отримання папки фотографа та її вмісту за шляхом та глибиною
IDropboxAccount Service	GetFile(ApplicationUser currUser, int fileId)	Отримання файлу фотографа за ідентифікатором
IDropboxAccount Service	GetPreviousFolders(Application User currUser, string folderpath, bool isPathToFolder = true)	Отримання попередніх папок вказаної папки (файлу) за шляхом
IDropboxAccount Service	GetAllFoldersWithGrantedUsers(ApplicationUser currUser)	Отримання всіх папок разом із клієнтами, які мають до них доступ

Продовження таблиці 2.2

Інтерфейс/Клас	Метод	Опис
IDropboxAccountService	GetSelectListItemOfAllFolders(ApplicationUser currUser)	Отримання списку всіх папок фотографа
IDropboxAccountService	GetFoldersGrantedToClient(ApplicationUser currUser)	Отримання списку папок до яких має доступ даний клієнт
IDropboxAccountService	GetFolderGrantedToClient(ApplicationUser currUser, int folderId)	Отримання папки до якої має доступ даний клієнт
IDropboxAccountService	GetClientFile(ApplicationUser currUser, int fileId)	Отримання файлу клієнта до якого він має доступ за ідентифікатором
IDropboxAccountService	GetFolderId(int fileId)	Отримання ідентифікатора папки за ідентифікатором файлу, який знаходиться у цій папці
IDropboxAccountService	SyncDropboxAccountContents(ApplicationUser currUser)	Виконання синхронізації з акаунтом Dropbox
IDropboxAccountService	CheckIfSyncWithDropboxIsNeeded(ApplicationUser currUser)	Перевірка чи потрібна синхронізація з акаунтом Dropbox

Продовження таблиці 2.2

Інтерфейс/Клас	Метод	Опис
IGrantAccessService	GrantAccessToFolders(GrantAccessDTO grantAccessDTO)	Виконання надання доступу до папок, використовуючи спеціальний клас-обгортку
DropboxAccountService	TraverseRecursive(DropboxClient dbx, string filepath, DropboxFolder folder, ApplicationUser currUser)	Рекурсивна функція додавання папки та файлів, що знаходяться в ній, до бази даних
IGrantAccessService	RevokeGrantToFolder(ApplicationUser currUser, string folderPath, string userEmail)	Виконання відкликання доступу до папки за шляхом та поштою клієнта
ILoginService	LoginUser(LoginViewModel user)	Отримання файлу клієнта до якого він має доступ за ідентифікатором
LoginService	VerifyCorrectPasswordAndEmail(ApplicationUser user, string password)	Перевірка паролю даного користувача

Продовження таблиці 2.2

Інтерфейс/Клас	Метод	Опис
IProfileService	GetRegistrationType(Application User user)	Отримання типу зареєстрованого користувача (клієнт, фотограф)
IProfileService	GetSelectListItemOfAllUsers(ApplicationUser currUser)	Отримання списку всіх користувачів системи окрім поточного
IRegistrationService	RegisterUser(RegisterViewModel user)	Виконання реєстрацію користувача у системі
IRegistrationService	DropboxAuthenticationSecondStep(ApplicationUser user, string code, string state, string redirectUri)	Перевірка автентичності відповіді від Dropbox при реєстрації та отримання токена для доступу до акаунту Dropbox (DropboxAccessToken)

Таблиця 2.3 – Опис класів виключних ситуацій.

Клас	Опис
ContentNotFoundException	Клас виключної ситуації: помилка при отриманні папки або файлу

Продовження таблиці 2.3

Клас	Опис
DropboxFolderEmptyException	Клас виключної ситуації: папка порожня
GrantAccessToFolderException	Клас виключної ситуації: помилка при наданні доступу до папки
NoAccessToContentException	Клас виключної ситуації: помилка при доступу до папки або файлу (наприклад, клієнту не було надано доступ до даної папки або файлу)
NoFoldersGrantedToClientException	Клас виключної ситуації: відсутність папок, доступних даному клієнту
SyncNeededException	Клас виключної ситуації: локальні дані не є актуальними по рівнянні з акаунтом Dropbox

Алгоритм синхронізації вмісту акаунту Dropbox

Синхронізація з акаунтом Dropbox є важливою частиною функціоналу даного програмного продукту. Припустимо, що фотограф змінив вміст свого акаунту Dropbox з клієнтського додатку Dropbox. Причиною зміни можуть бути цілком робочі моменти. Наприклад, фотограф виконав фотографування та завантажив фотографії у свій акаунт, фотограф перейменував, додав або видалив певні файли або папки і так далі. Для зберігання актуальності фотографій повинен бути розроблений алгоритм, який визначає зміни, які були виконані у акаунті Dropbox та певним чином змінює локальний стан папок. Словесний опис алгоритму синхронізації виглядає таким чином:

1. Отримання курсору поточної папки для порівняння із курсором у базі даних.
2. Якщо курсор змінився перехід до пункту 2.1, інакше до пункту 3.
 - 2.1. Отримання змін кореневої папки, шляхом запиту на
https://api.dropboxapi.com/2/file_requests/list_v2 та
https://api.dropboxapi.com/2/file_requests/list/continue.
 - 2.2. Ітерація по колекції змін серед елементів класу DeletedMetadata.
 - 2.2.1. Якщо це файл перехід до пункту 2.2.1.1, інакше це папка і перехід до пункту 2.2.1.2.
 - 2.2.1.1. Отримання поля DropboxId файлу у базі, якщо об'єкт класу FileMetadata відсутній, то це означає, що файл було просто видалено, виконання видалення файлу у базі даних. Інакше перехід до пункту 2.2.1.1.1.
 - 2.2.1.1.1. Якщо об'єкт класу FileMetadata з таким самим DropboxId відсутній, то це означає, що файл було просто видалено, виконання видалення файлу у базі даних. Інакше це перейменований файл, виконання перейменування файлу.
 - 2.2.1.2. Отримання поля DropboxId папки у базі, якщо об'єкт класу FolderMetadata відсутній, то це означає, що папку було просто видалено, виконання видалення папки у базі даних. Інакше перехід до пункту 2.2.1.2.1.
 - 2.2.1.2.1. Якщо об'єкт класу FolderMetadata з таким самим DropboxId відсутній, то це означає, що папку було просто видалено, виконання видалення папки у базі даних. Інакше це перейменована папка, виконання перейменування папки.
 - 2.3. Ітерація по колекції підпапок поточної папки.
 - 2.3.1. Перехід до пункту 1 з параметром – підпапка.

2.4. Ітерація по колекції змін серед елементів класу FileMetadata.

2.4.1. Так як це доданий файл – виконання додавання файлу у базу даних.

2.5. Ітерація по колекції змін серед елементів класу FolderMetadata.

2.5.1. Так як це додана папка – виконання додавання папки у базу даних.

3. Перехід до пункту 1 з параметром – підпапка.

Опис алгоритму синхронізації однієї папки у вигляді блок-схеми наведений на рисунку 2.8. При синхронізації всіх папок виконується рекурсивний виклик даного алгоритму.

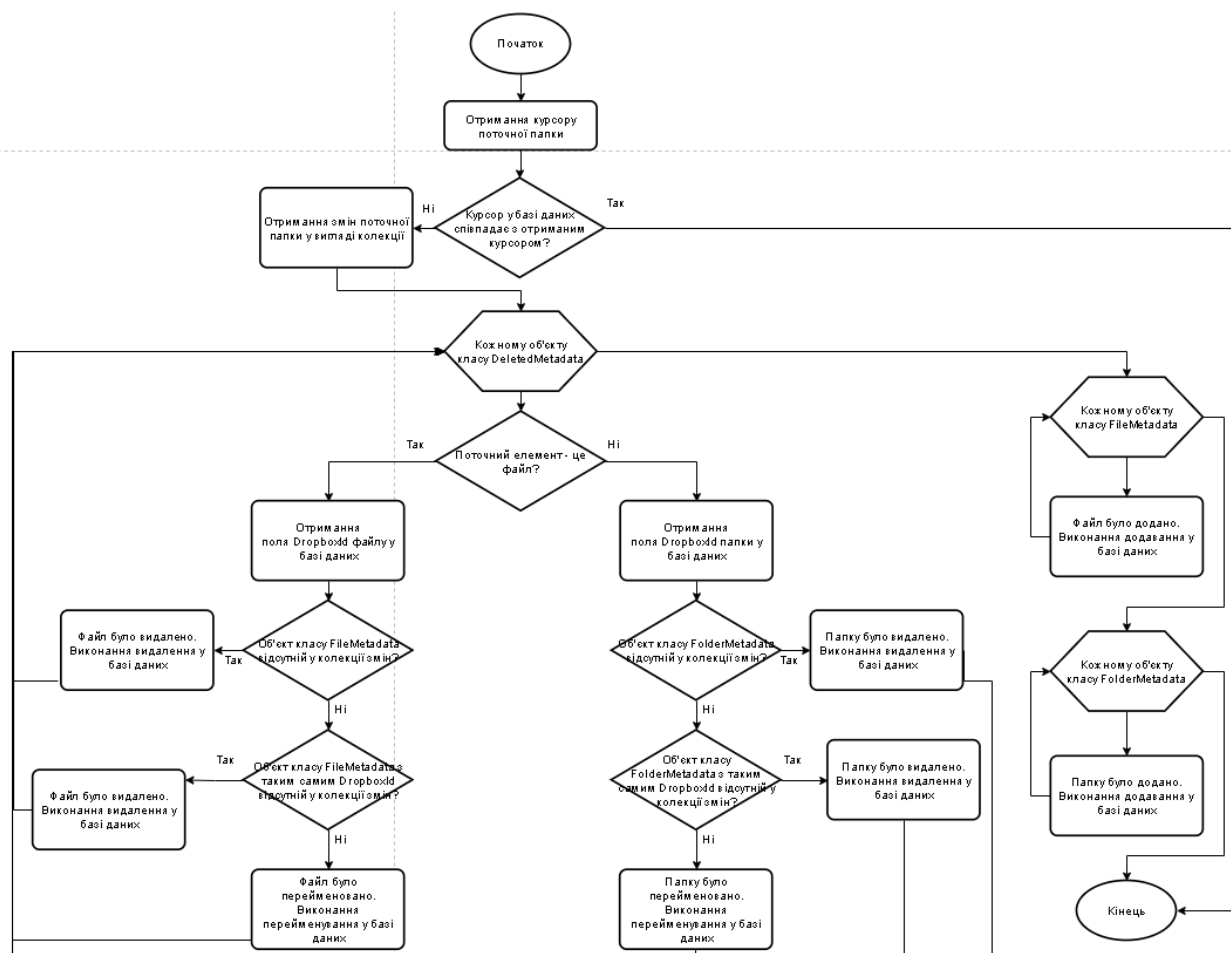


Рисунок 2.8 – Алгоритм синхронізації однієї папки

2.4 Аналіз безпеки даних

Для збереження паролів користувачів у базі даних було обрано метод використання KDF функції PBKFD2 (Password-Based Key Derivation Function 2). Ця функція налаштована на використання пароля користувача, випадкової солі, алгоритму HMAC-SHA256 і кількості ітерацій рівною 1000, які повинен застосовуватися HMAC. Використання цієї KDF функції обумовлено властивістю, якою вона володіє: значне уповільнення отримання паролю з хешу, який зберігається у базі даних. Тому у разі викриття бази даних такий метод зберігання паролів захищає від bruteforce атак, райдужних таблиць для розшифрування паролів користувачів.

2.5 Висновки до розділу

У даному розділі були розроблені функціональні та нефункціональні вимоги до програмного продукту, спроектовані схеми бізнес-процесів у застосунку, обрана мова програмування та фреймворки для написання сервісу та проаналізовано безпеку даних.

3 АНАЛІЗ ЯКОСТІ ТА ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

3.1 Аналіз якості ПЗ

Тестування являє собою невід’ємну частину розробки сучасних програмних продуктів. Для веб сервісів, які застосовують сторонні API, цей етап є ще важливішим. Перевірка коректності отримання даних з API, її обробка являють основну задачу при тестуванні. Ціллю процесу тестування є забезпечення якості ПЗ та наявності найбільш малої кількості багів, адже їх наявність у великих кількостях може негативно вплинути на відношення користувача до застосунку.

Загалом, при тестуванні повинні бути взяті до уваги наступні пункти:

- перевірка робота всіх обов’язкових функцій сайту;
- тестування працездатності форм на сайті, призначених для користувача (у тому числі оброблення виключних ситуацій);
- перевірка відображення фотографій;
- перевірка загального вигляду візуальної частини, правильність розташування блоків;
- перевірка правильності роботи з Dropbox API.

Тестування веб застосунку для електронної видачі фотозамовлень включає в себе такі види:

- компонентне тестування: націлено на забезпечення повноти та правильності логіки програми та забезпечення того, що окремий компонент працює як задумано. Серед модулів, які слід протестувати компонентним тестуванням можна виділити: модуль інтеграції з Dropbox, модуль збереження даних користувача у системі (реєстрація, авторизація), модуль

надання та відкриття доступу до папок, модуль відображення папок та файлів;

- інтеграційне тестування: націлено на забезпечення того, що модулі у поєднанні видають очікуваний результат і програмний продукт працює як єдине ціле. Серед модулів, які слід протестувати інтеграційним тестуванням можна виділити: отримання даних з Dropbox API та виведення їх на сайт;

- тестування інтерфейсу: націлено на забезпечення коректності полей введення даних, відображення блоків інформації та роботи користувацьких форм;

- приймальне тестування: націлено на забезпечення того, що продукт задовольняє та відповідає вимогам та очікуванням користувача. Серед таких тестів можна виділити: перевірка, що доступ до папок надається вказаним фотографом користувачам, перевірка, що відкриття доступу виконується для вказаних користувачів та папок.

Тестові модулі:

- модуль реєстрації та авторизації користувача;
- модуль інтеграції локального акаунту із акаунтом Dropbox;
- модуль синхронізації локального акаунту із акаунтом Dropbox;
- модуль надання та відкриття доступу до папок;
- модуль відображення папок та їх вмісту (фотографій);
- модуль підтвердження локального акаунту (відсилання повідомлення на пошту користувача для підтвердження акаунту).

Критерії призупинення тестування

Якщо дані, отримані з Dropbox API не є коректними, Dropbox API є тимчасово не доступним, то тестування слід призупинити.

Критерії поновлення

Якщо критерії призупинення були вирішені, то тестування слід поновити.

Критерії проходження

Якщо Dropbox API показує очікуваний результат, система авторизації працює коректно, система надання доступу до папок не виявляє помилок, фотографії відображаються для обох типів користувачів, то тестування слід затвердити. Для кожного виду тестування, якщо хоча б один тест не був пройдений, тестування вважається проваленим.

3.2 Опис процесів тестування

Розроблені тест-кейси були розділені на дві групи: позитивні тест-кейси та негативні тест-кейси. Позитивні тест-кейси наведені у таблиці 3.1.

Таблиця 3.1 – Позитивні тест-кейси

Тест-кейс	Кроки для відтворення	Очікуваний результат
Реєстрація фотографа	Перейти у вкладку форми реєстрації, вибрати тип реєстрації «Photographer», заповнити всі поля та натиснути «Register». Заповнити дані акаунту Dropbox та натиснути «Увійти»	Фотограф авторизується у системі, його дані коректно зберігаються у базі даних

Продовження таблиці 3.1

Тест-кейс	Кроки для відтворення	Очікуваний результат
Реєстрація клієнта	Перейти у вкладку форми реєстрації, вибрати тип реєстрації «Client», заповнити всі поля та натиснути «Register».	Клієнт авторизується у системі, його дані коректно зберігаються у базі даних
Авторизація користувача	Перейти у вкладку форми входу, заповнити поля пошти та паролю, натиснути кнопку підтвердження	Користувач авторизується у системі
Перегляд доступних папок	Авторизуватися як клієнт, натиснути на посилання «Folders I have access to»	Відображаються папки, доступні від різних фотографів
Перегляд доступних файлів усередині папок	Авторизуватися як клієнт, натиснути на посилання «Folders I have access to», натиснути на посилання «Browse folder» навпроти однієї з доступних папок	Відображається збільшений розмір фотографії разом із її назвою

Продовження таблиці 3.1

Тест-кейс	Кроки для відтворення	Очікуваний результат
Коректне відображення та можливість переходу у папки акаунту Dropbox	Авторизуватися як фотограф, перейти у вкладку «Dropbox account», попереходити у вкладені папки та файли, натискаючи на посилання «Browse folder» та «Browse file»	Переконалися у коректності відображення та переходів між вкладеними папками та файлами
Перегляд профілю фотографа	Авторизуватися як фотограф, перейти у вкладку «Profile»	Відображається представлення із особистою інформацією фотографа та інформацією його акаунту Dropbox
Перегляд профілю клієнта	Авторизуватися як клієнт, перейти у вкладку «Profile»	Відображається представлення із особистою інформацією клієнта
Відображення папок при наданні доступу	Авторизуватися як фотограф, перейти у вкладку «Granting»	Відображається правильний список папок фотографа, які відповідають папкам у вкладці «Dropbox account»

Продовження таблиці 3.1

Тест-кейс	Кроки для відтворення	Очікуваний результат
Пошук користувача при наданні доступу	Авторизуватися як фотограф, перейти у вкладку «Granting», ввести у текстове поле пошту користувача, який зареєстрований у системі, натиснути кнопку «Find client»	Відбувається додавання пошти введеного клієнта у список пошт та з'являється відповідний блок
Надання доступу на папки	Авторизуватися як фотограф, перейти у вкладку «Granting», вибрати папки, додати кілька клієнтів, натиснути кнопку підтвердження	Відображається повідомлення про успішне надання доступу. У акаунтах відповідних клієнтів у вкладці «Folders I have access to» з'являються відповідні папки із їх вмістом
Відкликання доступу до папок певним клієнтам	Авторизуватися як фотограф, перейти у вкладку «Granted folders», натиснути на посилання навпроти клієнта доступ якому потрібно заборонити (для певної папки), після відображення модального вікна, натиснути на кнопку підтвердження	Модальне вікно відображає повідомлення про успішне відкликання доступу та зникають відповідні записи клієнтів на поточній сторінці

Продовження таблиці 3.1

Тест-кейс	Кроки для відтворення	Очікуваний результат
Синхронізація із акаунтом Dropbox	Авторизуватися як фотограф, перейти у вкладку «Sync dropbox», натиснути на кнопку підтвердження	Відображається список папок, які були синхронізовані з акаунтом Dropbox
Відправлення повідомлення про підтвердження акаунту на пошту	Перейти до вкладки «Regiser», заповнити особисті дані та вибрати опцію виду реєстрації – клієнт, натиснути кнопку підтвердження.	Відображається повідомлення про те, що був відправлений лист для підтвердження акаунту на пошту. Лист з'явився у поштовому сервісі.

Негативні тест-кейси наведені у таблиці 3.2.

Таблиця 3.2 – Негативні тест-кейси

Тест-кейс	Кроки для відтворення	Очікуваний результат
Введення у пошуковий рядок шляху до вкладок авторизованого користувача	Додати або замінити пошуковий рядок на «/Account/UserProfile»	Відображається представлення із формою для входу

Продовження таблиці 3.2

Тест-кейс	Кроки для відтворення	Очікуваний результат
Невідповідність між паролем та підтвердженням паролю у формі реєстрації	Перейти у вкладку «Register», заповнити всі поля та залишити поля «Password» та «Confirm password» різними, натиснути кнопку підтвердження	З'являється повідомлення про те, що ці два поля повинні бути однаковими
Реєстрація з поштою, на яку вже є зареєстрований користувач у системі	Перейти у вкладку «Register», заповнити всі поля, зокрема поле «Email» заповнити поштою, яка вже є зареєстрованою у системі, натиснути кнопку підтвердження	З'являється повідомлення про помилку (дублікатний поштовий адрес)
Авторизація з вказанням неправильного паролю	Перейти у вкладку «Login», коректно заповнити поле «Email» та некоректно поле «Password», натиснути кнопку підтвердження	Відображається повідомлення про помилку авторизації
Перегляд доступних папок не підтвердивши акаунт	Авторизуватися як клієнт, який не підтвердив свій акаунт, перейти до вкладки «Folders I have access to»	Відображення представлення із повідомленням про необхідність підтвердження акаунту для перегляду доступних папок

Продовження таблиці 3.2

Перегляд доступної папки, у якій відсутні файли	Авторизуватися як клієнт, перейти у вкладку «Folders I have access to», перейти до папки у атрибуті «Number of files» якої вказано 0	Відображення представлення із повідомленням про відсутність файлів у наданій папці
Надання повторного доступу до папки	Авторизуватися як фотограф, перейти у вкладку «Granting», вибрати папку та клієнта, якому вже було надано доступ до цієї папки, натиснути кнопку підтвердження	Відображається повідомлення про помилку (повторне надання доступу до папки)
Додавання пошти клієнта, яка вже є у списку при наданні доступу	Авторизуватися як фотограф, перейти у вкладку «Granting», вибрати папку та додати пошту клієнта, повторно натиснути «Find client» з тою ж поштою	Відображення повідомлення про дублікатного клієнта
Синхронізація з акаунтом Dropbox	Авторизуватися як фотограф, переконавшись, що акаунт синхронізований, перейти до вкладки «Sync dropbox», натиснути кнопку підтвердження	Відображення повідомлення про відсутність необхідності синхронізації з Dropbox, адже не було змін у акаунті Dropbox

При проведенні тестування, спираючись на тест-кейси наведені у таблицях 3.1 та 3.2 було визначено, що всі тести були пройдені та показали очікуваний результат.

3.3 Опис контрольного прикладу

Для опису контрольного прикладу було вибрано одну з найголовніших функціональностей даного програмного продукту, а саме надання доступу до папок.

Передумови: бути авторизованим у якості фотографа.

Опис кроків тестування:

- перейти до вкладки «Granting»;
- вибрати потрібні папки зі списку папок Dropbox;
- по чергово додати кілька клієнтів у список бажаних клієнтів;
- натиснути кнопку підтвердження.

Очікуваний результат:

- у профілях клієнтів, яким був наданий доступ, у вкладці «Folders I have access to» з'явилися обрані папки;
- у профілі фотографа у вкладці «Granted folders» з'явилися записи доданих на доступ папок та користувачів.

Таким чином, вхідними даними є: список папок та клієнтів, а вихідними даними: наданий доступ до вказаних папок обраним клієнтам.

4 ВПРОВАДЖЕННЯ ТА СУПРОВІД ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

4.1 Розгортання програмного забезпечення

Для розгортання веб застосунку потрібна машина Windows Server з Internet Information Services (IIS) та додаток Visual Studio 2017 або вище. Для способу розгортання можна вибрати з поміж двох варіантів:

- шляхом запуску локального серверу IIS Express, який запускається за замовчуванням при відкритті сайту через Visual Studio;
- шляхом запуску локального серверу Local IIS та створення нового сайту у диспетчері служб IIS.

Для подальшого супроводу даного веб застосунку бажано розгорнути сайт на сервері Microsoft Azure, що надає доступ до сайту з мережі інтернет та крім цього має докладні графіки та аналітику запитів, помилок серверу тощо.

4.2 Робота з програмним забезпеченням

Детальна інструкція з експлуатації веб застосунку наведена у документі «Керівництво користувача».

ВИСНОВКИ

Метою дипломного проекту була розробка автоматизованого сервісу для електронної видачі фотозамовлень. Для досягнення цієї мети було проаналізовано предметну область, пов'язану з видачою фотографій, проаналізовано програмні продукти конкуренти та виділено їх недостатки та переваги, розроблено функціональні вимоги та варіанти використання, протестовано програмний продукт.

У результаті було отримано веб сайт, який виділяється такими перевагами, як: інтеграція з сервісом для хмарного сховища Dropbox, відсутність потреби у будь-яких інших допоміжних платформах, зручність у використанні для клієнтів та фотографів.

					КПІ.ІП-5208.045440.01.81	Арк.
						64
Змн.	Арк.	№ докум.	Підпис	Дата		

ПЕРЕЛІК ПОСИЛАНЬ

- 1) Хмарне сховище [Електронний ресурс]. – 2019. – Режим доступу:
<https://aws.amazon.com/ru/what-is-cloud-storage/>.
- 2) Dropbox [Електронний ресурс]. – 2018. – Режим доступу:
<https://uk.wikipedia.org/wiki/Dropbox>.
- 3) Mega [Електронний ресурс]. – 2019. – Режим доступу:
<https://ru.wikipedia.org/wiki/Mega>.
- 4) MVC [Електронний ресурс]. – 2019. – Режим доступу:
<https://ru.wikipedia.org/wiki/Model-View-Controller>.
- 5) MVC [Електронний ресурс]. – 2014. – Режим доступу:
<https://habr.com/ru/post/215605/>.
- 6) Dependency Injection [Електронний ресурс]. – 2017. – Режим доступу: <https://itvdn.com/ru/blog/article/injection>.
- 7) Creating a Model [Електронний ресурс]. – 2019. – Режим доступу:
<https://docs.microsoft.com/en-us/ef/ef6/modeling/>.
- 8) Anatomy of an ASP.NET Identity PasswordHash [Електронний ресурс]. – 2017. – Режим доступу:
<https://www.blinkingcaret.com/2017/11/29/asp-net-identity-passwordhash/>.
- 9) Things you wanted to know about storing passwords but were afraid to ask [Електронний ресурс]. – 2017. – Режим доступу:
<https://www.blinkingcaret.com/2017/11/15/things-wanted-know-storing-passwords-afraid-ask/>.
- 10) Тестування [Електронний ресурс]. – 2012. – Режим доступу:
<https://freshdesign.ua/blog/technology/website-testing>.

Факультет інформатики та обчислювальної техніки
Кафедра автоматизованих систем обробки інформації і управління

“ЗАТВЕРДЖЕНО”

В.о. завідувача кафедри

_____ О.А. Павлов

“ ____ ” _____ 2019 р.

Веб-сервіс для електронної видачі фото замовлень

Опис програми

КПІ.ІП-5208.045440.02.13

“ПОГОДЖЕНО”

Керівник проекту:

_____ Є.А. Недашківський

Нормоконтроль:

_____ К.І. Ліщук

Виконавець:

_____ А.В. Куш

Київ – 2019 року

Тексти програмного коду

Веб-сервіс для електронної видачі фото замовлень

(Найменування програми (документа))

CD-R

(Вид носія даних)

12 арк, 493 кб.

(Обсяг програми (документа) , арк.,) Кб)

Київ - 2019

					КПІ.ІП-5208.045440.02.13	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		2

```

public interface IGrantAccessService
{
    ErrorSink<DuplicateGrantingError>
    GrantAccessToFolders(GrantAccessDTO grantAccessDTO);
    void RevokeGrantToFolder(ApplicationUser currUser, string folderPath,
    string userEmail);
}

public class GrantAccessService : IGrantAccessService
{
    private readonly ApplicationDbContext db;

    public GrantAccessService()
    {
        db =
        HttpContext.Current.GetOwinContext().Get<ApplicationDbContext>();
        db.Configuration.ProxyCreationEnabled = true;
        db.Configuration.LazyLoadingEnabled = true;
    }

    private void GrantAccessToFoldersToUser(GrantAccessToUserDTO
    grantAccessToUserDTO, ErrorSink<DuplicateGrantingError> errors)
    {
        var destUser = db.Users.FirstOrDefault(u => u.Email ==
        grantAccessToUserDTO.User);
        if(destUser == null)
        {
            throw new GrantAccessToFolderException("Something went wrong
            when granting access.");
        }

        foreach(var folderId in grantAccessToUserDTO.Folders)
        {
            var folder =
            grantAccessToUserDTO.CurrUser.PhotographerFolders.SingleOrDefault(f =>
            f.PhotographerFolderId == folderId);
            if(folder == null)
            {
                throw new GrantAccessToFolderException("Something went wrong
                when granting access.");
            }

            if (destUser.GrantedFolders.Contains(folder))

```

```

        {
            errors.AddNewError(new DuplicateGrantingError(destUser, folder));
        }
        else
        {
            destUser.GrantedFolders.Add(folder);
        }
    }

    db.SaveChanges();
}

public ErrorSink<DuplicateGrantingError>
GrantAccessToFolders(GrantAccessDTO grantAccessDTO)
{
    ErrorSink<DuplicateGrantingError> errors = new
    ErrorSink<DuplicateGrantingError>();
    foreach(var userEmail /*userId*/ in grantAccessDTO.Users)
    {
        GrantAccessToFoldersToUser(new GrantAccessToUserDTO { CurrUser
        = grantAccessDTO.CurrUser, Folders = grantAccessDTO.Folders, User =
        userEmail }, errors);
    }

    return errors;
}

public void RevokeGrantToFolder(ApplicationUser currUser, string
folderPath, string userEmail)
{
    // get folder by folderPath
    var folder = currUser.PhotographerFolders.SingleOrDefault(f => f.Path ==
folderPath);
    if(folder == null)
    {
        throw new ContentNotFoundException("Folder is not found in your
dropbox account.");
    }

    // get user by userEmail
    var user = db.Users.SingleOrDefault(u => u.Email == userEmail);
    if(user == null)
    {

```

```

        throw new ContentNotFoundException("Destination user not found.");
    }

    // remove
    user.GrantedFolders.Remove(folder);
    db.SaveChanges();
}
}

public interface ILoginService
{
    Task<SignInStatus> LoginUser(LoginViewModel user);
}

public class LoginService : ILoginService
{
    private ApplicationUserManager userManager;
    private ApplicationSignInManager signInManager;

    public LoginService(ApplicationUserManager userManager,
        ApplicationSignInManager signInManager)
    {
        this.userManager = userManager;
        this.signInManager = signInManager;
    }

    public async Task<SignInStatus> LoginUser(LoginViewModel user)
    {
        var appUser = await userManager.FindByEmailAsync(user.Email);
        if(!VerifyCorrectPasswordAndEmail(appUser, user.Password))
        {
            return SignInStatus.Failure;
        }
        var result = await
signInManager.PasswordSignInAsync(appUser.UserName, user.Password,
user.RememberMe, shouldLockout: false);
        return result;
    }

    private bool VerifyCorrectPasswordAndEmail(ApplicationUser user, string
password)
    {
        if (user == null)
        {

```

```

        return false;
    }
    PasswordHasher passwordHasher = new PasswordHasher();
    var passVerificationResult =
passwordHasher.VerifyHashedPassword(user.PasswordHash, password);
    if (passVerificationResult == PasswordVerificationResult.Success)
    {
        return true;
    }
    return false;
}
}

public interface IProfileService
{
    string GetRegistrationType(ApplicationUser user);
    IEnumerable<SelectListItem> GetSelectListItemOfAllUsers(ApplicationUser
currUser);
}

public class ProfileService : IProfileService
{
    private ApplicationUserManager userManager;

    public ProfileService(ApplicationUserManager userManager)
    {
        this.userManager = userManager;
    }

    public string GetRegistrationType(ApplicationUser user)
    {
        string regType = string.Empty;
        if (user.ClientInfo == null && user.PhotographerInfo != null)
        {
            regType = "Photographer";
        }
        else if (user.PhotographerInfo == null && user.ClientInfo != null)
        {
            regType = "Client";
        }
        else if (user.ClientInfo != null && user.PhotographerInfo != null)
        {
            regType = "Photographer and Client";
        }
    }
}

```

```

    }
    else
    {
        regType = "Nobody";
    }
    return regType;
}

public IEnumerable<SelectListItem>
GetSelectListItemOfAllUsers(ApplicationUser currUser)
{
    var allUsers = userManager.Users.Where(c => c.Id !=
currUser.Id).ToList();

    var selectListItemOfUsers = allUsers.Select(o => new SelectListItem
    {
        Text = $"{o.Name} {o.Surname} - {o.Email}",
        Value = o.Id
    });

    return selectListItemOfUsers;
}

public interface IRegistrationService
{
    Task<ApplicationUser> RegisterUser(RegisterViewModel user);
    Task<bool> DropboxAuthenticationSecondStep(ApplicationUser user, string
code, string state, string redirectUri);
}

public class RegistrationService : IRegistrationService
{
    private ApplicationUserManager userManager;
    private ApplicationSignInManager signInManager;
    private readonly ApplicationDbContext db;

    public RegistrationService(ApplicationUserManager userManager,
ApplicationSignInManager signInManager, ApplicationDbContext db)
    {
        this.userManager = userManager;
        this.signInManager = signInManager;
        this.db = db;
    }
}

```



```

    public async Task<bool>
DropboxAuthenticationSecondStep(ApplicationUser user, string code, string state,
string redirectUri)
    {
        try
        {
            if (user.PhotographerInfo.ConnectState != state)
            {
                return false;
            }

            var response = await DropboxOAuth2Helper.ProcessCodeFlowAsync(
                code,
                MvcApplication.AppKey,
                MvcApplication.AppSecret,
                redirectUri);

            user.PhotographerInfo.DropboxAccessToken = response.AccessToken;
            await userManager.UpdateAsync(user);

            await signInManager.SignInAsync(user, isPersistent: false,
rememberBrowser: false);

            return true;
        }
        catch (Exception)
        {
            return false;
        }
    }

    public async Task<ApplicationUser> RegisterUser(RegisterViewModel user)
    {
        switch (user.RegTypeEnum)
        {
            case RegTypeEnum.Client:
                var appUser = new ApplicationUser { Username = user.Username,
Name = user.Name, Surname = user.Surname, Email = user.Email };
                var result = await userManager.CreateAsync(appUser,
user.Password);
                if (result.Succeeded)
                {

```

```

        await userManager.AddToRoleAsync(appUser.Id, "client");
        var clientInfo = new ClientInfo { ApplicationUserId = appUser.Id,
ApplicationUser = appUser, ClientInteger = 5 };
        appUser.ClientInfo = clientInfo;
        await userManager.UpdateAsync(appUser);

        await signInManager.SignInAsync(appUser, isPersistent: false,
rememberBrowser: false);

        return appUser;
    }
    return null;
    case RegTypeEnum.Photographer:
        var appUser1 = new ApplicationUser { UserName = user.Username,
Name = user.Name, Surname = user.Surname, Email = user.Email };
        var result1 = await userManager.CreateAsync(appUser1,
user.Password);
        if (result1.Succeeded)
        {
            await userManager.AddToRoleAsync(appUser1.Id,
"photographer");
            var photographerInfo = new PhotographerInfo { ApplicationUserId
= appUser1.Id, ApplicationUser = appUser1, ConnectState =
Guid.NewGuid().ToString("N") };
            appUser1.PhotographerInfo = photographerInfo;
            await userManager.UpdateAsync(appUser1);

            await signInManager.SignInAsync(appUser1, isPersistent: false,
rememberBrowser: false);

            return appUser1;
        }
        return null;
    default:
        return null;
    }
}
}

public interface IDropboxAccountService
{
    Task<string> GetDropboxAccountInfo(ApplicationUser currUser);
}

```

```

Task<DropboxAccountFileSystem>
GetContentsOfDropboxAccount(ApplicationUser currUser);

DropboxAccountFileSystem GetDropboxAccountFileSystem();

Task AddDropboxContentsToDB(ApplicationUser currUser);

DropboxFolderGrantedToClient GetFolderGrantedToClient(int folderId);

DropboxFolder GetFolder(ApplicationUser currUser, string folderpath, int
depth);

Task<DropboxFile> GetFile(ApplicationUser currUser, int fileId);

List<DropboxFolder> GetPreviousFolders(ApplicationUser currUser, string
folderpath, bool isPathToFolder = true);

List<DropboxFolder> GetAllFoldersWithGrantedUsers(ApplicationUser
currUser);

IEnumerable<SelectListItem>
GetSelectListItemOfAllFolders(ApplicationUser currUser);

// for client

List<DropboxFolderGrantedToClient>
GetFoldersGrantedToClient(ApplicationUser currUser);

DropboxFolderGrantedToClient
GetFolderGrantedToClient(ApplicationUser currUser, int folderId);

Task<DropboxFileGrantedToClient> GetClientFile(ApplicationUser
currUser, int fileId);

int GetFolderId(int fileId);

// synchronizing dropbox account

Task<SyncDropboxAccount>
SyncDropboxAccountContents(ApplicationUser currUser);

Task<bool> CheckIfSyncWithDropboxIsNeeded(ApplicationUser
currUser);

```

```

    }

    public class DropboxAccountService : IDropboxAccountService
    {
        private DropboxAccountFileSystem dropboxFileSystem = new
DropboxAccountFileSystem();

        private readonly ApplicationDbContext db;
        private FileExtensions fileExtensionsHelper;

        public DropboxAccountService()
        {
            db =
HttpContext.Current.GetOwinContext().Get<ApplicationDbContext>();

            db.Configuration.ProxyCreationEnabled = true;
            db.Configuration.LazyLoadingEnabled = true;
            fileExtensionsHelper = new FileExtensions();
        }

        public DropboxAccountFileSystem GetDropboxAccountFileSystem()
        {
            if (dropboxFileSystem.RootFolder.Files.Count == 0 &&
dropboxFileSystem.RootFolder.Folders.Count == 0)
            {
                throw new DropboxFolderEmptyException("Your Dropbox account is
empty.");
            }
            else
            {
                return dropboxFileSystem;
            }
        }
    }

```

```

    }
}

public async Task<DropboxAccountFileSystem> /*Task<List<string>>*/
GetContentsOfDropboxAccount(ApplicationUser currUser)
{
    if (currUser.PhotographerInfo != null &&
currUser.PhotographerInfo.DropboxAccessToken != null)
    {
        using (var dbx = new
DropboxClient(currUser.PhotographerInfo.DropboxAccessToken))
        {
            var allFiles = new List<string>();
            //return await FindRecursive(dbx, string.Empty, allFiles);
            await FindRecursive(dbx, string.Empty,
dropboxFileSystem.RootFolder);
        }
    }

    return GetDropboxAccountFileSystem();
}

```

Факультет інформатики та обчислювальної техніки
Кафедра автоматизованих систем обробки інформації і управління

“ЗАТВЕРДЖЕНО”

В.о. завідувача кафедри

_____ О.А. Павлов

“ ____ ” _____ 2019 р.

Веб сервіс для електронної видачі фото замовлень

Технічне завдання

КПІ.ІП-5208.045440.03.91

“ПОГОДЖЕНО”

Керівник проекту:

_____ Є.А.Недашківський

Нормоконтроль:

_____ К.І. Ліщук

Виконавець:

_____ А.В. Куш

Київ – 2019 року

ЗМІСТ

1	НАЙМЕНУВАННЯ ТА ГАЛУЗЬ ЗАСТОСУВАННЯ.....	3
2	ПІДСТАВА ДЛЯ РОЗРОБКИ.....	4
3	ПРИЗНАЧЕННЯ РОЗРОБКИ	5
4	ВИМОГИ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	6
4.1	Вимоги до функціональних характеристик.....	6
4.2	Вимоги до надійності	7
4.3	Умови експлуатації	7
4.4	Вимоги до складу і параметрів технічних засобів	7
4.5	Вимоги до інформаційної та програмної сумісності	8
4.6	Вимоги до маркування та пакування.....	8
4.7	Вимоги до транспортування та зберігання	8
4.8	Спеціальні вимоги.....	8
5	ВИМОГИ ДО ПРОГРАМНОЇ ДОКУМЕНТАЦІЇ	9
6	СТАДІЇ І ЕТАПИ РОЗРОБКИ.....	10
7	ПОРЯДОК КОНТРОЛЮ ТА ПРИЙМАННЯ.....	11

1 НАЙМЕНУВАННЯ ТА ГАЛУЗЬ ЗАСТОСУВАННЯ

Назва розробки: веб-сервіс для електронної видачі фото замовлень.

Галузь застосування:

Наведене технічне завдання поширюється на розробку веб-сервісу для електронної видачі фото замовлень, який використовується для електронної видачі фото замовлень та призначена для фотографів та їх клієнтів.

					КПІ.ІП-5208.045440.03.91	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		3

2 ПІДСТАВА ДЛЯ РОЗРОБКИ

Підставою для розробки веб-сервісу для електронної видачі фото замовлень є завдання на дипломне проектування, затверджене кафедрою автоматизованих систем обробки інформації і управління Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського» (КПІ ім.Ігоря Сікорського).

					КПІ.ІП-5208.045440.03.91	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		4

3 ПРИЗНАЧЕННЯ РОЗРОБКИ

Розробка призначена для надання фотографам зручної платформи для видачі їх робіт та клієнтам єдиного сайту для перегляду фотографій доступних від фотографів.

Метою розробки є автоматизація процесу видачі фото замовлень та створення єдиної платформи для реалізації даної цілі.

					КПІ.ІП-5208.045440.03.91	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		5

4 ВИМОГИ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

4.1 Вимоги до функціональних характеристик

4.1.1 Програмне забезпечення повинно забезпечувати виконання наступних основних функцій:

4.1.1.1 Для користувача:

- перегляд доступних від фотографів папок;
- надання доступу до папок;
- відкликання доступу до папок;
- синхронізація з акаунтом Dropbox;
- перегляд вмісту акаунту Dropbox;
- інтеграція локального акаунту фотографа з його акаунтом Dropbox.

4.1.2 Розробку виконати на платформі Windows

4.1.3 Додаткові вимоги не передбачені

4.2 Вимоги до надійності

4.2.1 Передбачити контроль введення інформації.

4.2.2 Передбачити захист від некоректних дій користувача.

4.2.3 Забезпечити цілісність інформації в базі даних.

4.3 Умови експлуатації

4.3.1 Умови експлуатації згідно СанПін 2.2.2.542 – 96.

4.3.2 Обслуговування

4.3.3 Обслуговуючий персонал

Для даного програмного продукту обслуговуючий персонал не передбачений.

4.4 Вимоги до складу і параметрів технічних засобів

4.4.1 Програмне забезпечення повинно функціонувати на IBM-сумісних персональних комп'ютерах.

4.4.2 Мінімальна конфігурація технічних засобів:

Тип процесору x86_64.

Об'єм ОЗП 1024 Мб.

Клавіатура та миша.

ЖК-дисплей.

4.5 Вимоги до інформаційної та програмної сумісності

4.5.1 Програмне забезпечення повинно працювати під управлінням операційних систем сімейства Windows або Unix.

4.5.2 Вхідні дані повинні бути представлені в наступному форматі: акаунт Dropbox, список папок, список файлів, список клієнтів.

4.5.3 Результати повинні бути представлені в наступному форматі: список папок, список файлів, які можуть бути переглянуті.

4.5.4 Програмне забезпечення повинно <опис інтерфейсу (протоколу) з іншим програмним забезпеченням>.

Середовище розробки – Visual Studio 2017, мова програмування – C#

4.6 Вимоги до маркування та пакування

Вимоги до маркування та пакування не пред'являються.

4.7 Вимоги до транспортування та зберігання

Вимоги до транспортування та зберігання не пред'являються.

4.8 Спеціальні вимоги

Згенерувати установчу версію програмного забезпечення.

5 ВИМОГИ ДО ПРОГРАМНОЇ ДОКУМЕНТАЦІЇ

5.1 Програмні модулі, котрі розробляються, повинні бути задокументовані, тобто тексти програм повинні містити всі необхідні коментарі.

5.2 Програмне забезпечення повинно мати довідникову систему

5.3 У склад супроводжувальної документації повинні входити наступні документи:

5.3.1 Пояснювальна записка не менше ніж на 60 аркушах формату А4 (без додатків 5.3.2 - 5.3.6).

5.3.2 Технічне завдання.

5.3.3 Керівництво користувача.

5.3.4 Програма та методика тестування

5.4 Графічна частина повинна бути виконана на аркушах формату А3, котрі включаються у якості додатків до пояснювальної записки:

5.4.1 Креслення екранних форм.

5.4.2 Схема структурна варіантів використання.

5.4.3 Схема структурна бази даних.

6 СТАДІЇ І ЕТАПИ РОЗРОБКИ

№	Назва етапу	Строк,	Звітність
1.	Вивчення літератури за тематикою проекту		
2.	Розробка технічного завдання		Технічне завдання
3.	Аналіз вимог та уточнення специфікацій		Специфікації програмного забезпечення
4.	Проектування структури програмного забезпечення, проектування компонентів		Схема структурна програмного забезпечення та специфікація компонентів (діаграма класів, схема алгоритму ...)
5.	Програмна реалізація програмного забезпечення		Тексти програмного забезпечення
6.	Тестування програмного забезпечення		Тести, результати тестування
7.	Розробка матеріалів текстової частини проекту		Пояснювальн а записка.
8.	Розробка матеріалів графічної частини проекту		Графічний матеріал проекту
9.	Оформлення технічної документації проекту		Технічна документація

7 ПОРЯДОК КОНТРОЛЮ ТА ПРИЙМАННЯ

7.1 Види випробувань

Тестування розробленого програмного продукту виконується відповідно до “Програми та методики тестування”.

					КПІ.ІП-5208.045440.03.91	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		11

Факультет інформатики та обчислювальної техніки
Кафедра автоматизованих систем обробки інформації і управління

“ЗАТВЕРДЖЕНО”

В.о. завідувача кафедри

_____ О.А. Павлов

“ ____ ” _____ 2019 р.

Веб-сервіс для електронної видачі фото замовлень

Програма та методика тестування

КПІ.ІІ-5208.045440.04.51

“ПОГОДЖЕНО”

Керівник проекту:

_____ Є.А. Недашківський

Нормоконтроль:

_____ К.І. Ліщук

Виконавець:

_____ А.В. Куш

Київ – 2019 року

ЗМІСТ

1	ОБ’ЄКТ ВИПРОБУВАНЬ	3
2	МЕТА ТЕСТУВАННЯ	4
3	МЕТОДИ ТЕСТУВАННЯ	5
4	ЗАСОБИ ТА ПОРЯДОК ТЕСТУВАННЯ	6

1 ОБ'ЄКТ ВИПРОБУВАНЬ

Об'єктом тестування є сервіс для електронної видачі фото замовлень, який є веб-сайтом створеним на платформі Microsoft .Net з використанням інфраструктури ASP.NET MVC.

					КПІ.ІП-5208.045440.04.51	Арк.
						3
Змн.	Арк.	№ докум.	Підпис	Дата		

2 МЕТА ТЕСТУВАННЯ

Метою тестування є перевірка наступних пунктів:

- коректна робота всіх сторінок веб-сервісу;
- коректна робота з Dropbox API;
- коректна робота системи реєстрації та авторизації;
- коректна робота системи надання та відкликання доступу до фотографій;
- відповідність програмних засобів вимогам технічного завдання.

					КПІ.ІП-5208.045440.04.51	Арк.
						4
Змн.	Арк.	№ докум.	Підпис	Дата		

3 МЕТОДИ ТЕСТУВАННЯ

Для тестування даного програмного продукту застосовуються наступні методи тестування:

- компонентне тестування;
- інтеграційне тестування;
- тестування інтерфейсу;
- приймальне тестування.

					КПІ.ІП-5208.045440.04.51	Арк.
						5
Змн.	Арк.	№ докум.	Підпис	Дата		

4 ЗАСОБИ ТА ПОРЯДОК ТЕСТУВАННЯ

Тестування проводилося у таких веб-браузерах, як: Mozilla Firefox, Google Chrome, Опера. Порядок тестування виглядає таким чином: компонентне тестування, інтеграційне тестування, тестування інтерфейсу, приймальне тестування.

Надійність веб-сайту перевіряється такими методами:

- ручне тестування, шляхом введення коректних значень у поля для вводу;
- ручне тестування, шляхом введення невалідних значень у поля для вводу;
- ручне тестування на відповідність функціональним вимогам;
- юніт-тести коду;
- тестування інтерфейсу веб-сайту.

					КПІ.ІП-5208.045440.04.51	Арк.
						6
Змн.	Арк.	№ докум.	Підпис	Дата		

Факультет інформатики та обчислювальної техніки
Кафедра автоматизованих систем обробки інформації і управління

“ЗАТВЕРДЖЕНО”

В.о. завідувача кафедри

_____ О.А. Павлов

“ ____ ” _____ 2019 р.

Веб-сервіс для електронної видачі фото замовлень

Схеми структурні

КПІ.ІП-5208.045440.05.99

“ПОГОДЖЕНО”

Керівник проекту:

_____ Є.А.Недашківський

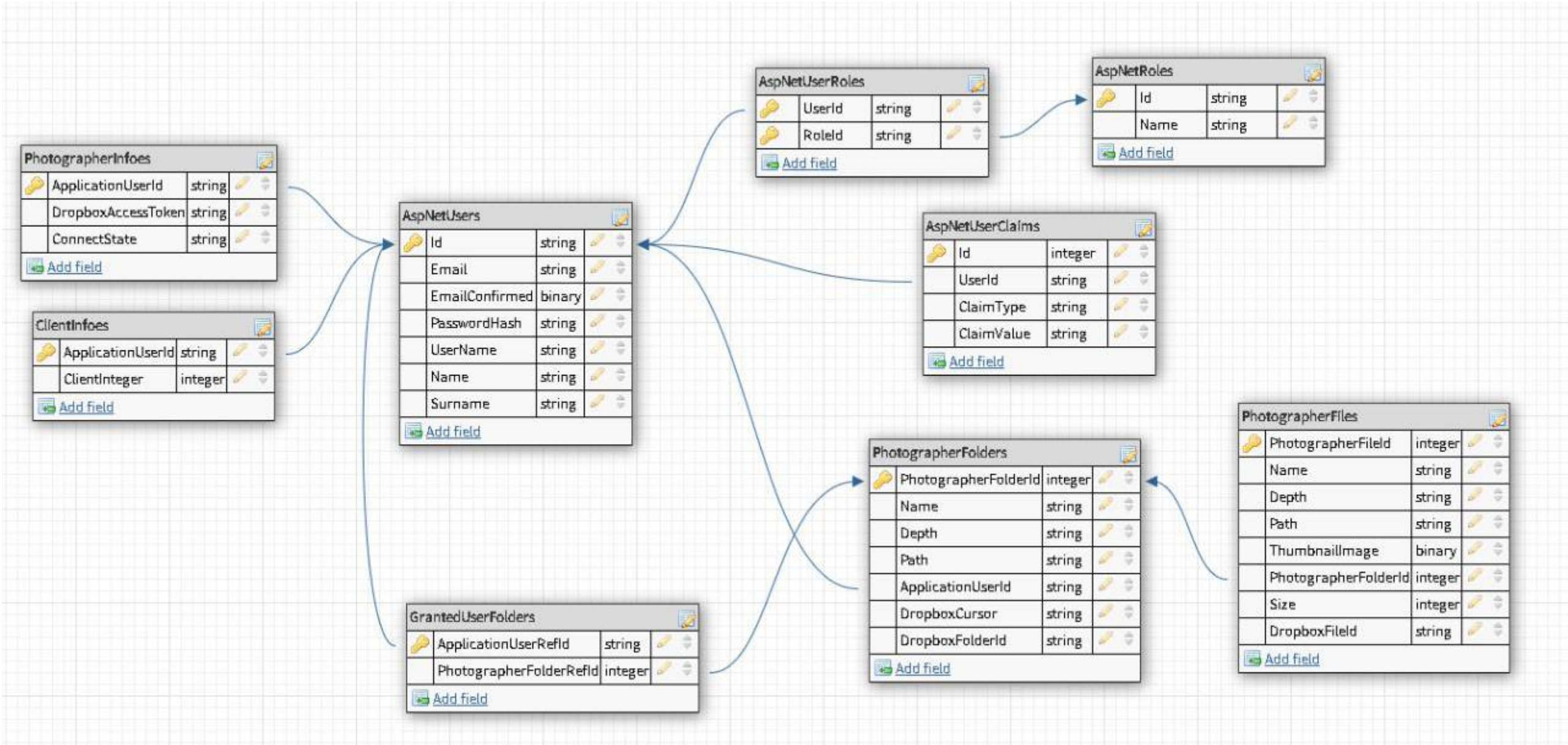
Нормоконтроль:

_____ К.І. Ліщук

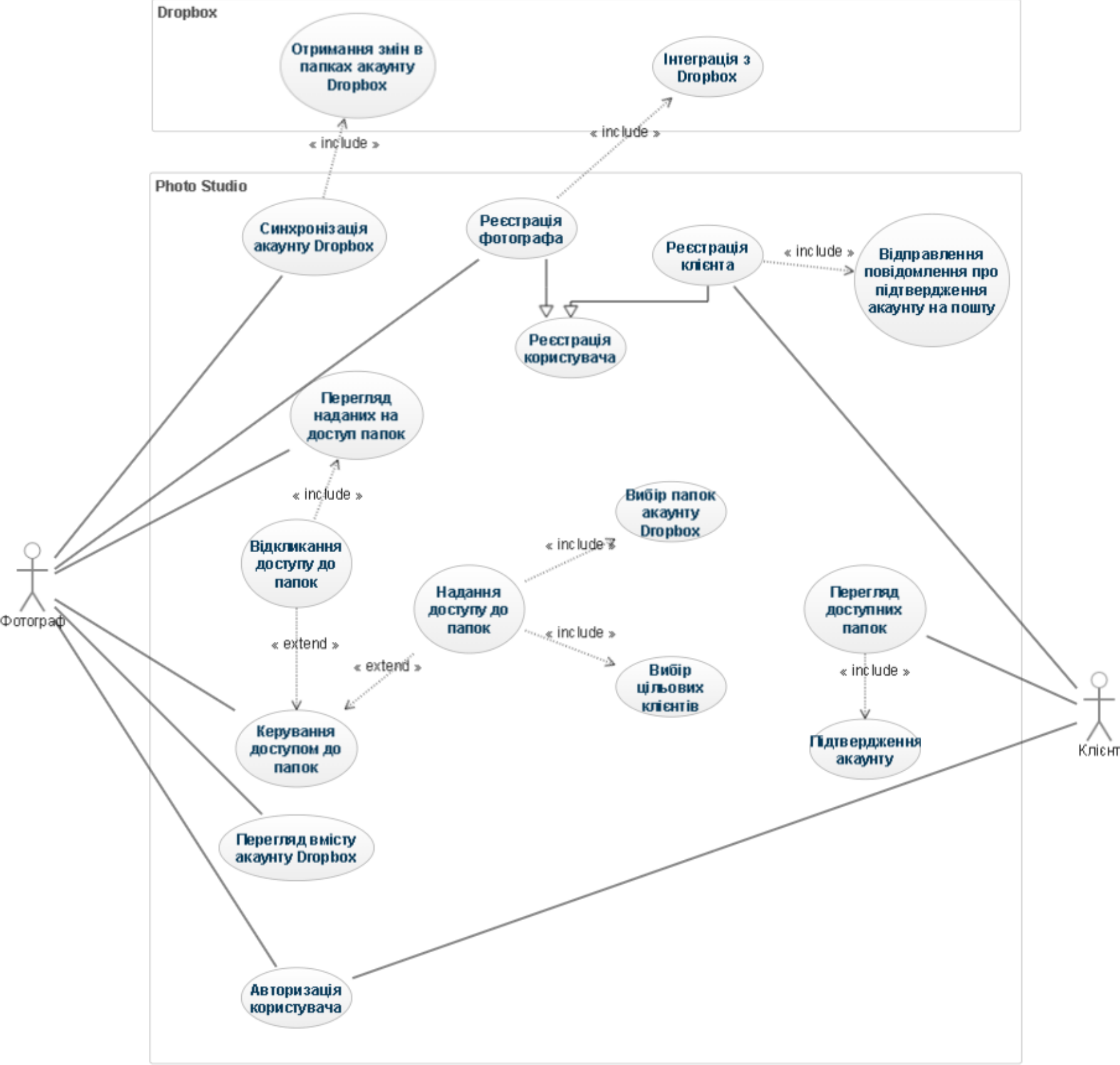
Виконавець:

_____ А.В.Куш

Київ – 2019 року



					КПІ.ІП-5208.045440.05.99.СБД			
					Схема бази даних	Літера	Маса	Масштаб
Зм.	Арк.	№ документа	Підпис	Дата				
Розробив		Куц А.В.						
Перевірів		Недашківський Є.А						
Т. кон.					Веб-сервіс для електронної видачі фото замовлень	Аркуш 1		Аркушів 1
Н. кон.		Ліщук К.І.				КПІ ім. Ігоря Сікорського кафедра АСОІУ гр. ІП-52		
Затвердив		Недашківський Є.А						



					КПІ.ІП-5208.045440.05.99.СС						
						Схема структурна варіантів використання	Літера			Маса	Масштаб
Зм.	Арк.	№ документа	Підпис	Дата	Схема структурна варіантів використання						
Розробив		Куц А.В.									
Перевірів		Недашківський Є.А.									
Т. кон.						Аркуш 1			Аркушів 1		
					Веб-сервіс для електронної видачі фото замовлень	КПІ ім. Ігоря Сікорського кафедра АСОІУ гр. ІП-52					
Н. кон.		Ліщук К.І.									
Затвердив		Недашківський Є.А.									

Photo Studio

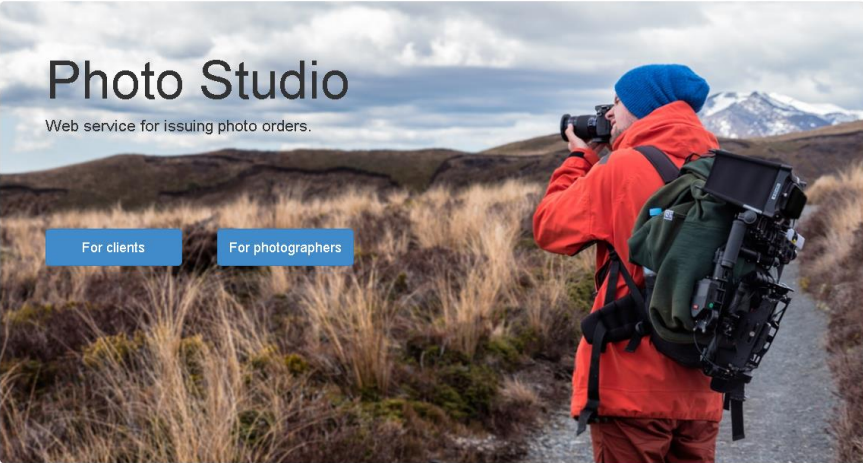
RegisterSign in

Photo Studio

Web service for issuing photo orders.

For clients

For photographers



For clients

Photo Studio

Dropbox accountGrantingGranted foldersSync Dropbox

Profile finalPhISign out

The contents of my Dropbox account

Folder > Folder CarrolFamilyPhotos >





ImageThumbnail	ContentName	IsFolder	Depth	FileSize	Goto
	spanishroyalfamily.jpg	<input type="checkbox"/>	1	40420	Browse file
	cinematographyShootBlack.jpg	<input type="checkbox"/>	1	121587	Browse file
	royal-family-christmas.jpg	<input type="checkbox"/>	1	5056913	Browse file
	houseBig.jpeg	<input type="checkbox"/>	1	47452	Browse file

Photo Studio

© 2019
All Rights Reserved

Photo Studio

Folders I have access to

Profile cliteeeee!Sign out

Folders that I have access to.





ImageThumbnail	ContentName	OwnerPhotographer	NumberOfFiles	Goto
	timeToStop	Kylian Mbappe - tarkov@gmail.com	4	Browse folder
	highwayRoutePass	Kylian Mbappe - tarkov@gmail.com	2	Browse folder
	CarrolFamilyPhotos	final final - finalph@gmail.com	4	Browse folder
	OwensFamilyPhotosSummer	final final - finalph@gmail.com	3	Browse folder

Photo Studio

© 2019
All Rights Reserved

Photo Studio

RegisterSign in

Register.

Create a new account.

Photographer

Client

Name

Surname

Username

Email

Password

Confirm password

Register

Photo Studio

© 2019
All Rights Reserved

Photo Studio

Dropbox accountGrantingGranted foldersSync Dropbox

Profile finalPhISign out

Dropbox file

Folder > Folder CarrolFamilyPhotos >

royal-family-christmas.jpg



Photo Studio

© 2019
All Rights Reserved

Photo Studio

RegisterSign in

Log in.

Use a local account to log in.

Email

Password

Log in

[Register as a new user](#)

Photo Studio

© 2019
All Rights Reserved

Photo Studio

Dropbox accountGrantingGranted foldersSync Dropbox

Profile finalPhISign out

Granted folders.

/carrollfamilyphotos

- tutorial@gmail.com [Revoke Grant](#)
- prisenko@gmail.com [Revoke Grant](#)

/owensfamilyphotossummer

- tutorial@gmail.com [Revoke Grant](#)
- prisenko@gmail.com [Revoke Grant](#)

Photo Studio

© 2019
All Rights Reserved

Granting access to specific users to specific folders

Select folders and users to give access to.

Select folder(s)

/carrollfamilyphotos, /owensfamilyphotossummer

Search user(s)

prisenko@gmail.com

Find user

tutorial@gmail.com

prisenko@gmail.com

Grant access

Photo Studio

© 2019
All Rights Reserved

					КПІ.ІП-5208.045440.05.99.KE					
					Креслення вигляду екранних форм	Літера			Маса	Масштаб
Зм.	Арк.	№ документа	Підпис	Дата						
Розробив	Куц А.В.									
Перевірив	Недашківський Є.А.									
Т. кон.							Аркуш 1			Аркушів 1
					Веб-сервіс для електронної видачі фото замовлень	КПІ ім. Ігоря Сікорського кафедра АСОІУ гр. ІП-52				
Н. кон.	Ліщук К.І.									
Затвердив	Недашківський Є.А.									

Факультет інформатики та обчислювальної техніки
Кафедра автоматизованих систем обробки інформації і управління

“ЗАТВЕРДЖЕНО”

В.о. завідувача кафедри

_____ О.А. Павлов

“ ____ ” _____ 2019 р.

Веб-сервіс для електронної видачі фото замовлень

Керівництво користувача

КПІ.ІП-5208.045440.06.34

“ПОГОДЖЕНО”

Керівник проекту:

_____ Є.А.Недашківський

Нормоконтроль:

_____ К.І. Ліщук

Виконавець:

_____ А.В.Куш

Київ – 2019 року

При відкритті веб сайту у браузері користувач потрапляє на головну сторінку зображену на рисунку 1.1.

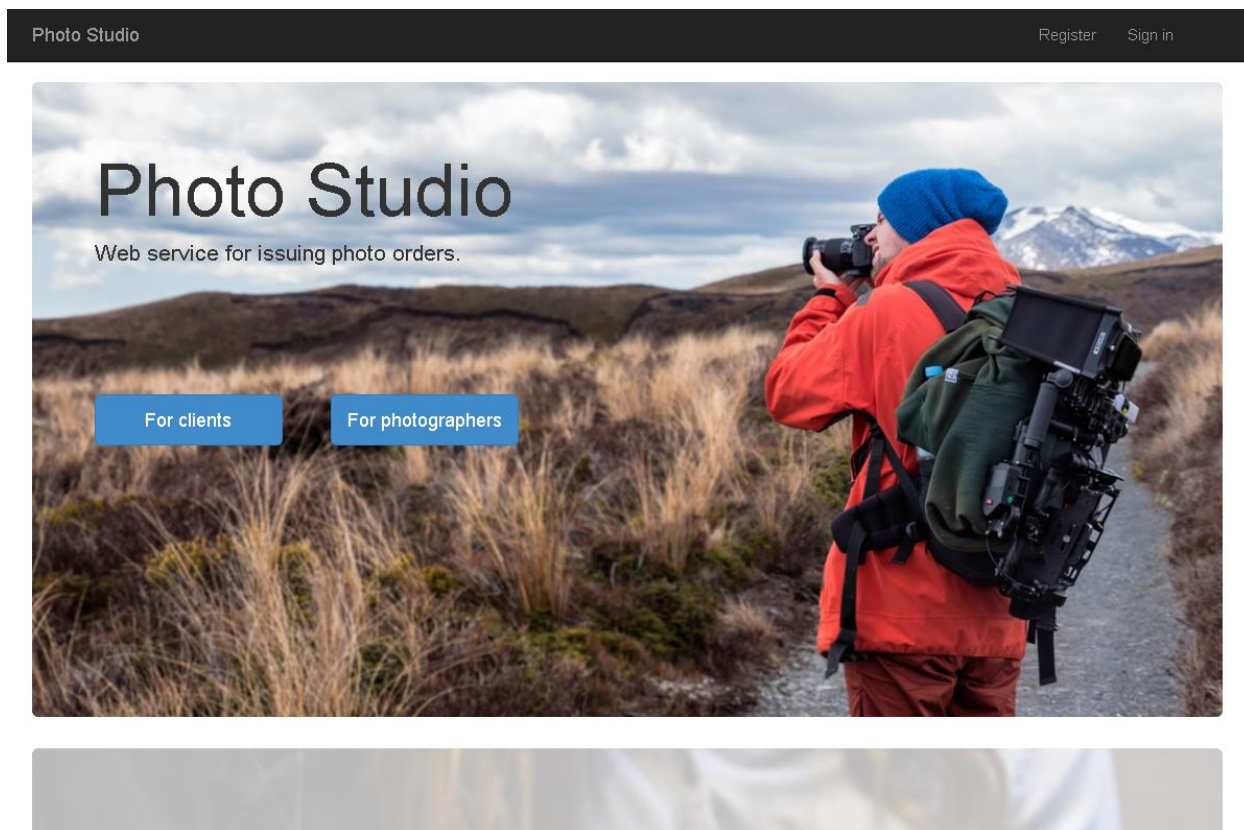


Рисунок 1.1 – Головна сторінка

Оглянувши коротке призначення даного сервісу та переваги для фотографів та клієнтів, користувач може перейти до вкладки «Register». Вміст цієї вкладки зображений на рисунку 1.2.

					КПІ.ІП-5208.045440.06.34	Арк.
						2
Змн.	Арк.	№ докум.	Підпис	Дата		

Register.

Create a new account.

☒ Photographer

☐ Client

Name

Surname

Username

Email

Password

Confirm password

Register

Рисунок 1.2 – Форма реєстрації

Заповнивши особисті дані, вибравши тип реєстрації «фотограф» та натиснувши кнопку підтвердження система валідує введені дані та у разі успіху перенаправляє користувача на форму інтеграції з його акаунтом Dropbox. Дана форма наведена на рисунку 1.3.

Змн.	Арк.	№ докум.	Підпис	Дата



Чтобы подсоединиться к infopulse, войдите в Dropbox.



Войти через Google

или

Адрес электронной почты

Пароль

Эта страница защищена функцией геСАРТСНА и подпадает под [Политику конфиденциальности](#) и [Условия использования](#) Google.

[Забыли пароль?](#)

Войти

Только начинаете знакомство с Dropbox? [Создать аккаунт](#)

Рисунок 1.3 – Форма інтеграції з Dropbox

Після введення логіну та паролю система пропонує фотографу погодитися з правами доступу до його акаунту Dropbox.



Luka Modric



infopulse хочет получить доступ к файлам и папкам в вашем Dropbox. [Дополнительные сведения](#)

Отмена

Можно

Рисунок 1.4 – Погодження з правами доступу до Dropbox

					КПІ.ІП-5208.045440.06.34	Арк.
						4
Змн.	Арк.	№ докум.	Підпис	Дата		

Далі система авторизує фотографа. Вигляд екрану браузеру зображений на рисунку 1.5.

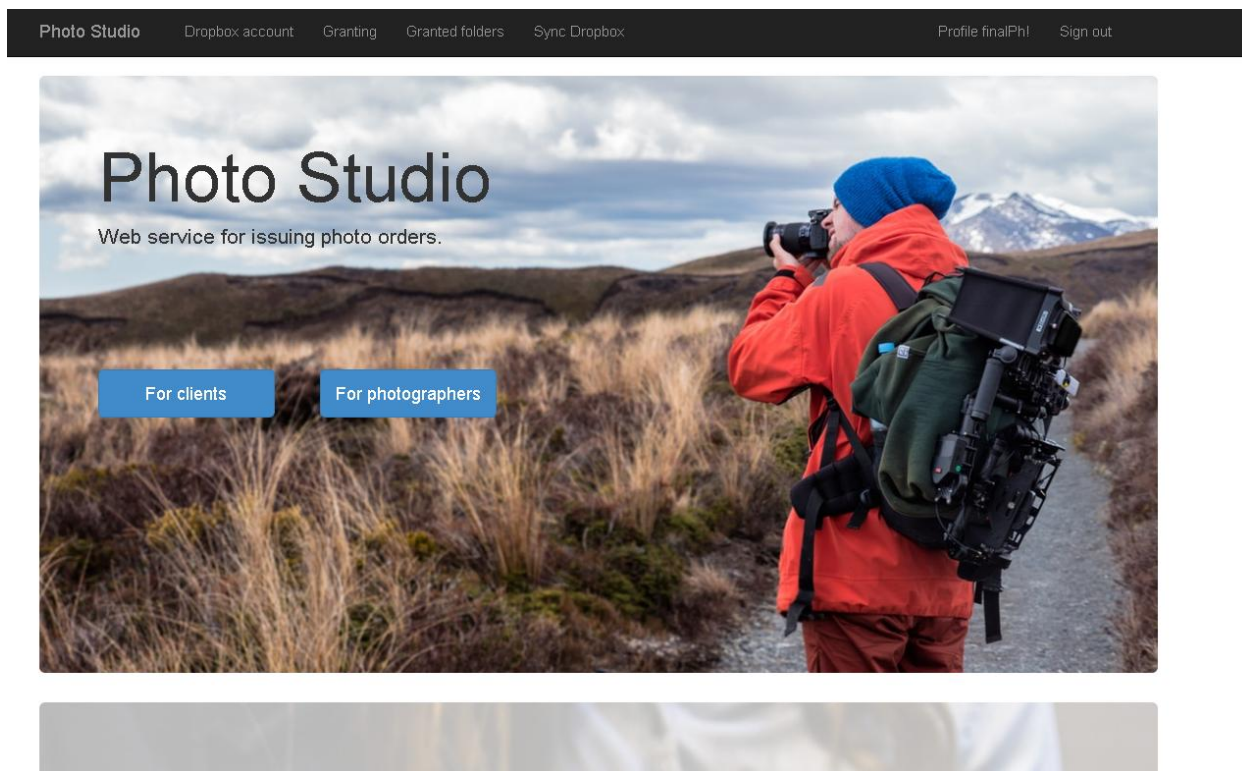


Рисунок 1.5 – Екран авторизованого фотографа

Далі фотограф може переглянути вміст свого акаунту Dropbox, перейшовши до вкладки «Dropbox account». Вміст екрану зображений на рисунку 1.6.

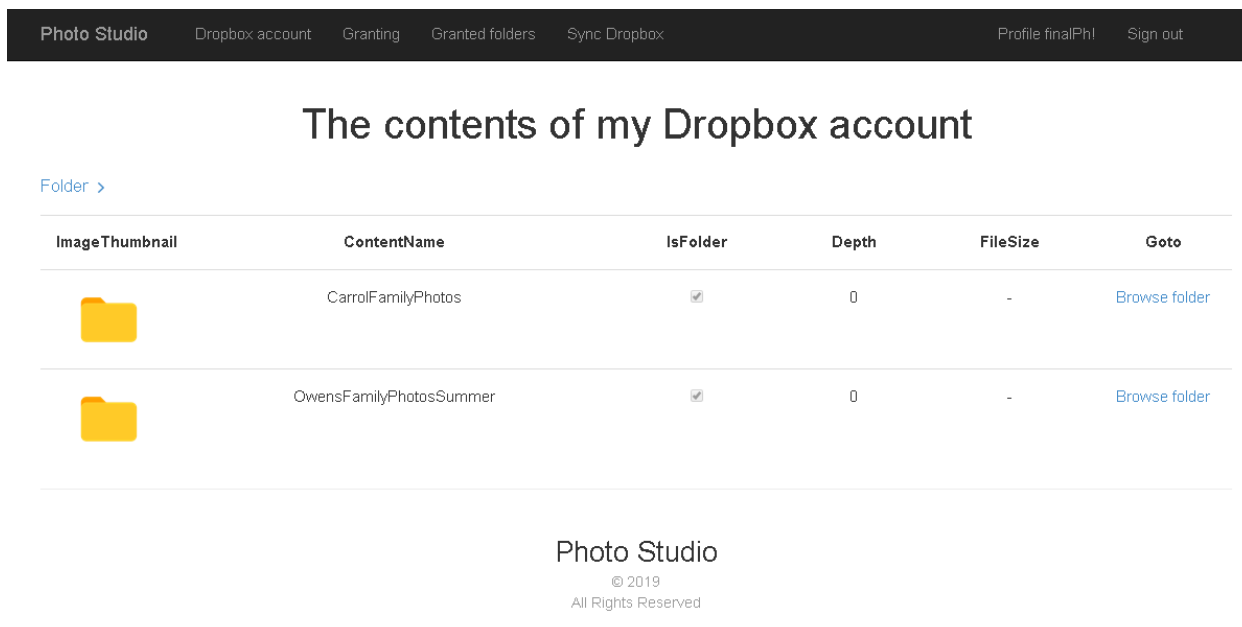


Рисунок 1.6 – Вміст акаунту Dropbox

Щоб переглядати папки та файли необхідно натискати на посилання у колонці «Goto», посилання мають назви «Browse folder» та «Browse file».

Перейшовши до вкладки «Granting» відображається форма для надання доступу до папок, у якій фотограф вибирає папки та клієнтів, яким він бажає надати доступ. Вміст вкладки «Granting» зображений на рисунку 1.7.

Рисунок 1.7 – Вкладка «Granting»

Для вибору папок необхідно вибрати бажані зі списку, який при натисканні розкривається. Для вибору клієнтів необхідно ввести пошту клієнта у текстове поле та натиснути кнопку «Find user». Таким чином можна додавати необхідних клієнтів.

Перейшовши до вкладки «Granted folders» відображається список папок та користувачів, які вже мають до них доступ. Вміст екрану зображений на рисунку 1.8.

Granted folders.

/carrollfamilyphotos

- tutorial@gmail.com [Revoke Grant](#)
- prisenko@gmail.com [Revoke Grant](#)

/owensfamilyphotossummer

- tutorial@gmail.com [Revoke Grant](#)
- prisenko@gmail.com [Revoke Grant](#)

Photo Studio

© 2019

All Rights Reserved

Рисунок 1.8 – Вкладка «Granted folders»

Натискаючи на посилання, яке знаходиться у відповідному рядку клієнта, фотограф може відкликати доступ до певної папки певному клієнту. Перед відкликанням доступу відображається модальне вікно для підтвердження цієї дії.

Перейшовши до вкладки «Sync Dropbox» відображається представлення для синхронізації з акаунтом Dropbox, зображене на рисунку 1.9.

Synchronize Dropbox account contents

Press the button to synchronize.

Photo Studio

© 2019

All Rights Reserved

Рисунок 1.9 – Вкладка «Sync Dropbox»

Для виконання синхронізації необхідно натиснути на кнопку підтвердження.

					КПІ.ІП-5208.045440.06.34	Арк.
						7
Змн.	Арк.	№ докум.	Підпис	Дата		

Для виходу з профілю необхідно натиснути на кнопку «Sign out».

Для реєстрації у якості клієнта у формі реєстрації потрібно вибрати опцію реєстрації для клієнта. Після підтвердження реєстрації буде відображено повідомлення про те, що на пошту клієнта було відправлено повідомлення для підтвердження акаунту. Для доступу до папок, які були надані фотографами необхідно підтвердити свій акаунт.

Якщо клієнт підтвердив свій акаунт перешовши за посиланням вказаним у повідомленні, яке було надіслано поштою, у вкладці «Folders I have access to» будуть відображатися папки, які були надані фотографами.

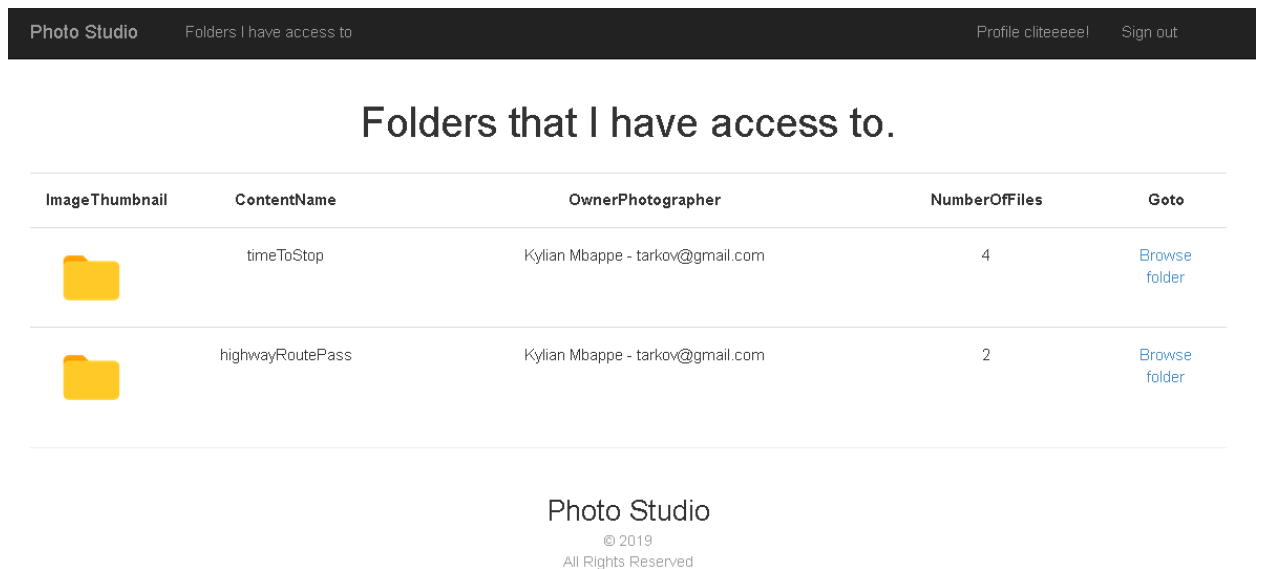


Рисунок 1.11 – Папки, доступні від фотографів